60000288

CONTROL DATA

CYBER 96X MSL MAINTENANCE SOFTWARE REFERENCE MANUAL

MDT3/P TEST DESCRIPTION

CDC® COMPUTER SYSTEMS
CYBER 960, 962

PREFACE

This manual describes the CDC CYBER 960 and 962 Computer Systems MDT3/P test. Procedures for using this test are provided in the CYBER 96X Test Procedures Reference manual.

NOTE

It is assumed and planned that the user of program descriptions will have an in-depth understanding of the processor logic; including familiarity with all levels of the Multilevel Block Diagrams.

DISCLAIMER

This product is intended for use only as described in this document. Control Data cannot be responsible for the proper functioning of undescribed features or undefined parameters.

REVISION RECORD

REVISION	DESCRIPTION
A (07-11-88)	Manual released.

Publication No.
60000288

REVISION LETTERS I, O, Q, S, X, AND Z ARE NOT USED

1988

by Control Data Corporation
Printed in the United States of America

Address comments concerning this manual to:

Control Data Corporation
Technology and Graphics Division
4201 Lexington Avenue North
St. Paul, Minnesota 55126-6198

or use Comment Sheet in the back of this manual.

CONTENTS

General 1
Section Descriptions 5
Section 0 - Checks of ALN Soft Control Memory Addressing and Minor Cycle Control Register 5
Section 0, Subsection 0 8
Section 0, Subsection 1 11
Section 0, Subsection 2 13
Section 1 - Checks of C Register Entry Controls and Sign Extension .. 16
Section 1, Subsection 1 21
Section 1, Subsection 2 24
Section 1, Subsection 3 26
Section 2 - Basic Checks of Multiply Logic 28
Section 2, Subsection 0 35
Section 2, Subsection 1 37
Section 2, Subsection 2 41
Section 2, Subsection 3 44
Section 2, Subsection 4 46
Section 2, Subsection 5 48
Section 2, Subsection 6 50
Section 2, Subsection 7 52
Section 2, Subsection 8 54
Section 2, Subsection 9 56
Section 2, Subsection 10 57
Section 2, Subsection 11 59
Section 2, Subsection 12 61
Section 2, Subsection 13 63
Section 2, Subsection 14 65
Section 2, Subsection 15 68
Section 2, Subsection 16 70
Section 2, Subsection 17 73
Section 2, Subsection 18 75
Section 2, Subsection 19 79
Section 2, Subsection 20 83
Section 2, Subsection 21 86
Section 2, Subsection 22 89
Section 2, Subsection 23 93
Section 2, Subsection 24 96
Section 2, Subsection 25 99
Section 2, Subsection 26 102
Section 2, Subsection 27 105
Section 2, Subsection 28 109
Section 2, Subsection 29 112
Section 2, Subsection 30 115
Section 2, Subsection 31 118
Section 2, Subsection 32 121
Section 2, Subsection 33 124
Section 2, Subsection 34 127
Section 2, Subsection 35 130

Section 2, Subsection 36 133
Section 2, Subsection 37 136
Section 2, Subsection 38 139
Section 2, Subsection 39 142
Section 3 - Check Multiply Final Adder Logic 145
Section 3, Subsection 0 150
Section 3, Subsection 1 152
Section 3, Subsection 2 153
Section 3, Subsection 3 156
Section 3, Subsection 4 158
Section 3, Subsection 5 162
Section 3, Subsection 6 164
Section 4 - Check B Register Entry Controls 167
Section 4, Subsection 0 170
Section 4, Subsection 1 173
Section 4, Subsection 2 175
Section 4, Subsection 3 179
Section 5 - Check C Register Entry and 170 Multiply Controls 182
Section 5, Subsection 0 185
Section 5, Subsection 1 187
Section 5, Subsection 2 189
Section 5, Subsection 3 191
Section 6 - Checks of C Register Entry, Multiply/Divide Control Sign Signals to the ALN General Network and the Major Cycle Control Buffer 196
Section 6, Subsection 0 199
Section 6, Subsection 1 201
Section 6, Subsection 2 205
Section 6, Subsection 3 209
Section 6, Subsection 4 213
Section 6, Subsection 5 217
Section 6, Subsection 6 219
Section 6, Subsection 7 221
Section 6, Subsection 8 223
Section 7 - Verify Operation of the Divide Quotient Register 226
Section 7, Subsection 0 229
Section 7, Subsection 1 231
Section 7, Subsection 2 234
Section 7, Subsection 3 236
Section 7, Subsection 4 237
Section 7, Subsection 5 241
Section 8 - Checks of Divisor and Remainder Register Loading 243
Section 8, Subsection 0 247
Section 8, Subsection 1 250
Section 8, Subsection 2 253
Section 8, Subsection 3 255
Section 8, Subsection 4 258
Section 8, Subsection 5 260
Section 9 - Check Internal Divide Array Logic 264
Section 9, Subsection 0 268
Section 9, Subsection 1 275
Section 9, Subsection 2 282

Section 10 - Check Integer Divide Operation and Pop Counter 284
Section 10, Subsection 0 287
Section 10, Subsection 1 289
Section 10, Subsection 2 294
Section 11 - Check Multiply/Divide Unit for Error Status Reporting, Parity Checking and Parity Generation 297
Section 11, Subsection 0 300
Section 11, Subsection 1 302
Section 11, Subsection 2 304
Section 11, Subsection 3 305
Section 11, Subsection 4 307
Section 11, Subsection 5 309

APPENDIXES

A. GLOSSARY A-1

B. MDT3/P MICRAND DESCRIPTIONS B-1

THIS PAGE HAS INTENTIONALLY BEEN LEFT BLANK

MULTIPLY/DIVIDE UNIT TEST MDT3/P

==========

GENERAL

MDT3/P tests the Multiply/Divide Unit logic block and provides error detection of all hardware failures. Failure data, when analyzed, will aid with hardware isolation.

The diagnostic consists of 12 sections. The following defines the major areas tested by these sections.

- o Section 0 checks the ALN soft control memory addressing and the Minor Cycle Control register.
- o Section 1 checks the C register.
- o Sections 2 through 3 perform an exhaustive check of all the basic multiply logic.
- o Sections 4 through 6 perform an exhaustive check of all multiply control and status logic.
- o Sections 7 through 10 perform an exhaustive check of the divide logic and Pop Counter.
- o Section 11 performs an exhaustive check of the parity logic.

All processor activity (micrand execution) is controlled from a previously tested peripheral processing unit (PPU). Controlling signals are sent to and received from processor via the Maintenance Channel (MCH) and the processor Maintenance Access Control (MAC).

A typical test sequence would be:

1. Load processor registers with operand patterns.
2. Send processor a Start command.
3. Processor now executes the diagnostic micrand or micrand sequence and halts processor.
4. The PPU senses halt and directs MAC, via PPU resident code, to read test results from the processor registers.
5. The data retrieved is examined for failure.

SECTION/SUBSECTION ORDER DEPENDENCIES:

MDT3/P is organized into section, subsection and condition elements. The successful completion of any element is prerequisite on the successful completion of all its predecessor elements; therefore, the sequential order of execution is encouraged.

Failures detected when test elements are executed out of order should not be used for hardware isolation analysis.

LOGIC TERMINOLOGY:

The processor logic circuit terminology in this manual has been updated to match the terminology used in publications level 1 through 4 diagrams. Circuit terminology used in the MDT3/P test code documentation (PP code and processor microcode) has not been updated. When in doubt about which terminology is correct (manual or test code), the manual terminology should be assumed correct.

TEST INITIALIZATION:

- Load control store.
- Load address control soft control memory with a captured product set version.
- Load ALN soft control with first soft control overlay.
- Clear User Mask register.
- Clear Monitor Mask register.
- Clear 170 Exit Mode Mask register.
- Clear the Processor Test Mode register.
- Clear Monitor Condition register.
- Clear Central Memory Bounds register.
- Clear Central Memory DEC register, except for the timing bits, which are not altered.
- Initialize processor DEC register as:
  - Set RMA mode bit.
  - Set Disable PDMs bit.
  - Do not change First Failure Capture enable bit.
  - Do not change timing bits.
  - Clear all other bits.
- Set VMID to 180 mode.
- Clear the Trap Enable register.
- Write zeros in RF0B and RF0C.
- Write the following ALN soft control memory map.

Address	32-Bit Value to be Written (Hexadecimal)
0	7920 0100
1	0300 0000
2	0108 0000
3	0000 0000
4	0308 7920
5	4108 0320
6	0108 0108
7	0000 0000

- Run Microand Sequence 68 to initialize the Multiply/Divide Unit Quotient and C Registers to zero.

NOTE

These conditions, once initialized, are reinstated at the completion of every section that alters them except for ALN Soft Control Memory. It is, therefore, assumed that these conditions exist at the beginning of every section. This is true even when section selections other than the default selections are run.

REGISTER FILE USE:

Register	Use
RF06	Live register write data.
RF07	Scratch or input register for two or more microand sequences.
RF08	Scratch or input register for two or more microand sequences.
RF09	Input or result register when more than two inputs are requested or when two results are required.
RF0A	Final result register.
RF0B	Input register for microand sequences.
RF0C	Input register for microand sequences.
RF0D	Result register for live register read of the UCR.

TRAPS AND NORMAL HALT:

All hardware-forced traps, should they occur during processor microcode execution, will spin at the trap address and set the central processing unit (CPU) halted status. No traps are expected in this test. Normal halt is always at control store address (CSA) 003(HEX), which sets the CPU halted status and spins at that address.

ALN MICRAND FIELDS:

Refer to Functional Unit Microands in the CYBER 960 and 962 Theory and Diagrams Manual (publication number 60000118)for detailed descriptions of the ALN microand fields.

NOTE

It is assumed and planned that the user of this documentation has an in-depth understanding of the processor logic, including familiarity with all levels of the processor multilevel diagrams.

REPEAT ITERATION:

Except where otherwise noted, setting Repeat Iteration (bit 2**13 in PARAM1) will cause the repetition of all conditions in a subsection.

SECTION DESCRIPTIONS

SECTION 0 - CHECKS OF ALN SOFT CONTROL MEMORY ADDRESSING AND MINOR CYCLE CONTROL REGISTER

This section verifies the following areas of the multiply/divide logic: addressing of the soft control memory from the ALN micrand field and minor cycle clock, the Minor Cycle Control register clock enables from soft control memory bits or ALN idle state, the Minor Cycle Control register parity logic and Minor Cycle Control register parity error reporting from ALN micrand auxiliary board to the Processor Fault Status register.

Subsection 0 - Test the Minor Cycle Control register for parity status in Idle state, parity error detection and reporting, and clock enable controls from soft control memory.

Subsection 1 - Test soft control memory addressing from ALN micrand field and minor cycle clock.

Subsection 2 - Test Minor Cycle Control register parity checkers and all bit paths into the register.

Method:

Refer to Subsection Method for details.

All subsections in this section will read and format the following:

- Processor halt address.
- Processor halt status.
- ALN micrand parity status.
- Soft Control register parity status.

Section Status Format:

The following bit assignments are used by all conditions that report status.

NOTE

Any status not read by a condition and all unused bit positions will be reported as zero. Refer to Section Method for status that will be read and reported.

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
0	Soft Control Register Parity Error byte 0	PFSALN(62)	PFS88/14
1	Soft Control Register Parity Error byte 1	PFSALN(63)	PFS88/15
2	Not Used		
3	Not Used		
4	Not Used		
5	Not Used		
6	Multiply Carry Miscompare byte 0	PFSALN(44)	PFS88/60
7	Multiply Carry Miscompare byte 1	PFSALN(45)	PFS88/61
8	Not Used		
9	Multiply Carry Miscompare byte 2	PFSALN(46)	PFS89/0
10	Multiply Carry Miscompare byte 3	PFSALN(47)	PFS89/1
11	Multiply Carry Miscompare byte 4	PFSALN(48)	PFS89/2
12	Multiply Carry Miscompare byte 5	PFSALN(49)	PFS89/3
13	Multiply Carry Miscompare byte 6	PFSALN(50)	PFS89/4
14	Multiply Carry Miscompare byte 7	PFSALN(51)	PFS89/5
15	Multiply Carry Miscompare byte 8	PFSALN(52)	PFS89/6
16	Multiply C Register Parity Error byte 0	PFSALN(0)	PFS88/16
17	Multiply C Register Parity Error byte 1	PFSALN(1)	PFS88/17
18	Multiply C Register Parity Error byte 2	PFSALN(2)	PFS88/18
19	Multiply C Register Parity Error byte 3	PFSALN(3)	PFS88/19
20	Multiply C Register Parity Error byte 4	PFSALN(4)	PFS88/20
21	Multiply C Register Parity Error byte 5	PFSALN(5)	PFS88/21
22	Multiply C Register Parity Error byte 6	PFSALN(6)	PFS88/22
23	Multiply C Register Parity Error byte 7	PFSALN(7)	PFS88/23
24	Multiply B Register Parity Error byte 0	PFSALN(8)	PFS88/24
25	Multiply B Register Parity Error byte 1	PFSALN(9)	PFS88/25
26	Multiply B Register Parity Error byte 2	PFSALN(10)	PFS88/26

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
27	Multiply B Register Parity Error byte 3	PFSALN(11)	PFS88/27
28	Multiply B Register Parity Error byte 4	PFSALN(12)	PFS88/28
29	Multiply B Register Parity Error byte 5	PFSALN(13)	PFS88/29
30	Multiply B Register Parity Error byte 6	PFSALN(14)	PFS88/30
31	Multiply B Register Parity Error byte 7	PFSALN(15)	PFS88/31
32	DAI Data Parity Error byte 0	PFSOPI(27)	PFS84/48
33	DAI Data Parity Error byte 1	PFSOPI(28)	PFS84/49
34	DAI Data Parity Error byte 2	PFSOPI(29)	PFS84/50
35	DAI Data Parity Error byte 3	PFSOPI(30)	PFS84/51
36	DAI Data Parity Error byte 4	PFSOPI(31)	PFS84/52
37	DAI Data Parity Error byte 5	PFSOPI(32)	PFS84/53
38	DAI Data Parity Error byte 6	PFSOPI(33)	PFS84/54
39	DAI Data Parity Error byte 7	PFSOPI(34)	PFS84/55
40	ALN Micrand Parity Error byte 0	PFSALN(54)	PFS89/8
41	ALN Micrand Parity Error byte 1	PFSALN(55)	PFS89/9
42	ALN Micrand Parity Error byte 2	PFSALN(56)	PFS89/10
43	ALN Micrand Parity Error byte 3	PFSALN(57)	PFS89/11
44	ALN Micrand Parity Error byte 4	PFSALN(58)	PFS89/12
45	ALN Micrand Parity Error byte 5	PFSALN(59)	PFS89/13
46	ALN Micrand Parity Error byte 6	PFSALN(60)	PFS89/14
47	ALN Micrand Parity Error byte 7	PFSALN(61)	PFS89/15
48	Not Used		
49	Not Used		
50	Not Used		
51	Not Used		
52	Processor Did Not Halt Flag from PP Program		
53	Control Store Address bit 0		CSA/53
54	Control Store Address bit 1		CSA/54
55	Control Store Address bit 2		CSA/55
56	Control Store Address bit 3		CSA/56
57	Control Store Address bit 4		CSA/57
58	Control Store Address bit 5		CSA/58
59	Control Store Address bit 6		CSA/59
60	Control Store Address bit 7		CSA/60
61	Control Store Address bit 8		CSA/61
62	Control Store Address bit 9		CSA/62
63	Control Store Address bit 10		CSA/63

Section 0, Subsection 0

Test Minor Cycle Control register for parity status in idle state, parity error detection and reporting, and clock enable controls from soft control memory.

Hardware Tested:

- Idle state parity bits.
- Minor Cycle Control register parity error detection and reporting logic.
- ALN Micrand Aux for busy enable of soft control parity status reporting.
- Minor Cycle Control register clock enable selects.
- PFS register 88 bits 14 and 15.
- ALN micrand bit 55 selection of soft control memory bits.

Method:

The Minor Cycle Control register testing will be done by using the load enable control from soft control memory bits 7 and 23 to enable or disable its clock.

Idle state parity status is checked by executing a micrand that inhibits functional units (causing idle state to be loaded). A micrand is then issued to ALN with an ALN micrand that attempts to load soft control memory address zero, which contains all zeros, and will not load. ALN busy causes the minor cycle control register parity status to be gated to PFS register 88 bits 14 and 15 from which it will be read and reported.

Parity error detection and reporting and Minor Cycle Control register clock enables are tested by writing bad parity into the first location of soft control memory with and without the enable clock bit set. An ALN micrand will be executed that reads this location and enters either bits 0 through 15 or 16 through 31 into the Minor Cycle Control register under control of ALN micrand bit 55 and enable clock Minor Cycle Control register bits 7 and 23. Parity status from PFS register bits 14 and 15 will be reported and should equal the expected result.

Micrands Used:

Sequence Number	Description
1	Load ALN idle state and ALN soft control address zero into Minor Cycle Control register.
65	Load ALN idle state to clear previous error.
2	Load Minor Cycle Control register from ALN soft control entry point zero.
3	Load Minor Cycle Control register from ALN soft control entry point 10(HEX).

Conditions:

C.0 Test idle state parity status.
- a. Start processor at Micrand Sequence 1; load idle state.
- b. Wait halt.
- c. Read and report Minor Cycle Control register parity status, processor halted status and processor halt address according to Section Status Format.
- d. Error if not equal to 0000 0000 0000 0003(HEX).

C.1 Write bad parity in ALN soft control memory.
- a. Write 0100 0001(HEX) with bad parity in ALN soft control memory location 0.
- b. Report soft control memory location 0 in bytes 0 through 3; informational.

C.2 Test byte zero parity error detection.
- a. Start processor at Micrand Sequence 65; load ALN idle state to clear previous error.
- b. Start processor at Micrand Sequence 2; load ALN soft control entry point 0.
- c. Wait halt.
- d. Read and report Minor Cycle Control register parity status, processor halted status and processor halt address according to Section Status Format.
- e. Error if not equal to 8000 0000 0000 0003(HEX).

C.3 Write bad parity in ALN soft control memory.
- a. Write 0001 0100(HEX) in ALN soft control memory location 0.
- b. Report ALN soft control memory location 0 in bytes 0 through 3; informational.

C.4 Test loading from ALN soft control bits 16 through 31.
- a. Start processor at Micrand Sequence 65; load ALN idle state to clear previous error.
- b. Start Micrand Sequence 3; load from ALN soft control entry point 10(HEX).
- c. Wait halt.
- d. Read and report Minor Cycle Control register parity status, processor halted status and processor halt address.
- e. Error if not equal to 8000 0000 0000 0003(HEX).

C.5 Test block of clock enable when bit 7 of soft control is not present.
- a. Start processor at Micrand Sequence 65; load ALN idle state to clear previous error.
- b. Start processor at Micrand Sequence 2; load from ALN soft control entry point 0.
- c. Wait halt.
- d. Read and report Minor Cycle Control register parity status, processor halted status and processor halt address according to Section Status Format.
- e. Error if not equal to 0000 0000 0000 0003(HEX).

Conditions:

C.6 Write bad parity in ALN soft control.
- a. Write 1101 1000(HEX) with bad parity in ALN soft control location 0.
- b. Report ALN soft control location 0 in bytes 0 through 3; informational.

C.7 Test byte 1 parity error detection.
- a. Start processor at Micrand Sequence 65; load ALN idle state to clear previous error.
- b. Start processor at Micrand Sequence 2; load from ALN soft control entry point 0.
- c. Wait halt.
- d. Read and report Minor Cycle Control register parity status, processor halted status and processor halt address.
- e. Error if not equal to 4000 0000 0000 0003(HEX).

C.8 Test block of clock enable when bit 23 of soft control is not present.
- a. Start processor at Micrand Sequence 65; load ALN idle state to clear previous error.
- b. Start processor at Micrand Sequence 3; load from ALN soft control entry point 10(HEX).
- c. Wait halt.
- d. Read and report Minor Cycle Control register parity status, processor halted status and processor halt address according to Section Status Format.
- e. Error if not equal to 0000 0000 0000 0003(HEX).

C.9 Write bad parity in ALN soft control memory.
- a. Write 1000 1101(HEX) with bad parity in ALN soft control memory location 0.
- b. Report ALN soft control memory location 0 in bytes 0 through 3; informational.

C.10 Test byte 1 parity detection from bits 16 through 31 of ALN soft control memory.
- a. Start processor at Micrand Sequence 65; load ALN idle state to clear previous error.
- b. Start processor at Micrand Sequence 3; load from ALN soft control entry point 10(HEX).
- c. Wait halt.
- d. Read and report Minor Cycle Control register parity status, processor halted status and processor halt address according to Section Status Format.
- e. Error if not equal to 4000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
- a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

Write 0000 0000(HEX) in ALN soft control memory location 0.

Section 0, Subsection 1

This subsection tests soft control memory addressing from ALN micrand soft control entry point and minor cycle clock.

Hardware Tested:

- ALN Micrand soft control entry point into Address Mux B on ALN soft control memory board.
- Clock generation of the lower 2 bits of the ALN micrand soft control address on the ALN micrand aux board.

Method:

The ALN micrand field soft control entry point addressing of the ALN soft control memory is tested by writing a word with bad parity into the soft control memory location accessed by the ALN micrand field soft control entry point being tested and the parity status from the Minor Cycle Control register will confirm that the proper address was read.

The soft control test pattern will be written with bad parity into the soft control memory location specified by the test pattern. The test soft control entry point specified by the test pattern will be written into the soft control entry point field of ALN micrand portion of Micrand Sequence 4. The micrand is then executed and a parity error reported if the proper address was read.

The soft control location is then written with all zeros to remove the parity errors. The above procedure is repeated until all test patterns have been run.

Micrands Used:

Sequence Number	Description
4	Load Minor Cycle Control register from soft control entry point xx. (The soft control entry point is written into the micrand by the peripheral processor (PP) program.)
65	Load ALN idle state to clear previous error.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

Conditions:

C.1 Write ALN soft control memory and Micrand Sequence 4.
- a. Write the test pattern into the ALN soft control memory location specified by the iteration count with bad parity.
- b. Write the ALN micrand soft control address specified by the pattern number into Micrand Sequence 4 in control store.
- c. Report test pattern in bytes 0 through 3; informational.
- d. Report soft control location written into byte 5; informational.
- e. Report soft control entry point written into Micrand Sequence 4 in byte 7; informational.

C.2 Test addressing of soft control memory.
- a. Start processor at Micrand Sequence 65; load ALN idle state to clear previous error.
- b. Start processor at Micrand Sequence 4; load Minor Cycle Control register.
- c. Wait halt.
- d. Read and report Minor Cycle Control register parity status, processor halted status, and processor halt address according to Section Status Format.
- e. Error if not equal to C000 0000 0000 0003(HEX).
- f. Write 0000 0000(HEX) into soft control memory location specified by the test pattern.

C.3 Report First Failure parity information.
- a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.1.

Exit subsection if no more test patterns.

Test Patterns (Hexadecimal):

Iteration Count	Test Pattern	Location to be Written into Soft Control	Entry Point to be Written into Micrand 4
0	0101 0000	00	00
1	0101 0000	01	00
2	0101 0000	02	00
3	0101 0000	03	00
4	0101 0000	04	01
5	0101 0000	09	02
6	0101 0000	12	04

Iteration Count	Test Pattern	Location to be Written into Soft Control	Entry Point to be Written into Micrand 4
7	0101 0000	23	08
8	0101 0000	3C	0F
9	0101 0000	28	0A
A	0101 0000	14	05

Section 0, Subsection 2

This subsection tests Minor Cycle Control register parity checkers and all bit paths into the register.

Hardware Tested:

- Minor Cycle Control register.

Method:

The parity checkers in the Minor Cycle Control register are checked using a 0 through FF(HEX) incrementing pattern. This pattern also checks all the bit paths into the register.

The ALN soft control memory is fully written with the first overlay of the test pattern. The soft control entry point to be read is written into the ALN micrand portion of Micrand Sequence 4. Micrand Sequence 4 is executed and no parity errors are expected. The soft control entry point written into Micrand Sequence 4 is incremented from 0 through 1F and the micrand is executed for each value. Then the second overlay of the ascending pattern is written into the ALN soft control memory. The soft control entry point to be written into Micrand Sequence 4 is reset to 0, written into the sequence, the micrand executed, and the parity status checked. No parity errors are expected. The soft control entry point is again incremented through 1F, with Micrand Sequence 4 being executed for each address. No parity error status is expected.

Micrands Used:

Sequence Number	Description
4	Load Minor Cycle Control register from soft control entry point xx. (The soft control entry point is written into the micrand by the PP program.)

Initialization:

- Write overlay 1 into ALN soft control memory.

Conditions:

The following conditions are repeated until ALN micrand soft control entry point 1F has been executed on the second overlay of the ascending test pattern.

C.0
a. Report iteration count in byte 7; informational.
b. In byte 2, report a 1 if first overlay has been written, or a 2 if second overlay has been written; informational.

C.1 Update and write ALN micrand soft control entry point.
a. Get current entry point; ascends from 0 through 1F(HEX).
b. Write current entry point into Micrand Sequence 4.
c. Report current entry point; informational.

C.2 Test Minor Cycle Control register.
a. Start processor at Micrand Sequence 4; load Minor Cycle Control register.
b. Wait halt.
c. Read and report Minor Cycle Control register parity status, processor halted status, and processor halt address.
d. Error if not equal to 0000 0000 0000 0003(HEX).

C.3 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if current soft control entry point equals 1F and second overlay test pattern has been written into soft control memory.

Exit section and write initialization overlay into ALN soft control memory.

Test Patterns (Hexadecimal):

Iteration Count	Entry Point	SCM Address	First Overlay
00/10	00/10	00	0100 4140
		01	0101 4141
		02	0302 4342
		03	0303 4343
		04	0504 4544
01/11	01/11	05	0505 4545
.	.	.	.
.	.	.	.
0F/1F	0F/1F	3C	3D3D 7D7C
		3D	3D3D 7D7D
		3E	3F3E 7F7E
		3F	3F3F 7F7F
	Second Overlay		
20/30	00/01	00	C1C0 8180
		01	C1C1 8181
		02	C3C2 8382
		03	C3C3 8383
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.
2F/3F	0F/1F	3C	FDFC BDBC
		3D	FDFD BDBD
		3E	FFFE BFBE
		3F	FFFF BFBF

SECTION 1 - CHECKS OF C REGISTER ENTRY CONTROLS AND SIGN EXTENSION

This section verifies the operation of the C register, C sign selection logic, sign extension using the Divide Quotient register and C register entry controls.

Subsection 0 - Test C register select of shifted data and sign extension from bit 0.

Subsection 1 - Test 32-bit sign extension in divide Quotient register and C register entry from Divide Quotient register.

Subsection 2 - Test C sign selection logic under 170/180 mode micrand control and its entry into the C register.

Subsection 3 - Test unsigned C register entry from major and minor cycle controls.

Method:

- Hardware testing. Refer to Subsection Method for details.

Status reporting:

All conditions that report status in this section will read and format the following status according to the section status format.

- ALN micrand bytes 0 through 7 parity status.
- DAI bytes 0 through 7 parity status.
- Minor Cycle Control register bytes 0 and 1 parity status.
- Processor halted status.
- Processor halt address; CSA.

Initialization:

- Done as Part of subsection 0 PP code.
- Write following ALN Soft Control Memory Map into location specified.

Location	Value to be Written
0	4108 5108
1	C108 C108
2	0108 0108
3	0000 0000
4	0000 5188
5	0000 C108
6	0000 0108
7	0000 0000

- Write zeros in RFOB.

Section Status Format:

The following bit assignments are used by all conditions that report status.

NOTE

Any status not read by a condition and all unused bit positions will be reported as zero. Refer to Section Method for status that will be read and reported.

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
0	Soft Control Register Parity Error byte 0	PFSALN(62)	PFS88/14
1	Soft Control Register Parity Error byte 1	PFSALN(63)	PFS88/15
2	Not Used		
3	Not Used		
4	Not Used		
5	Not Used		
6	Multiply Carry Miscompare byte 0	PFSALN(44)	PFS88/60
7	Multiply Carry Miscompare byte 1	PFSALN(45)	PFS88/61
8	Not Used		
9	Multiply Carry Miscompare byte 2	PFSALN(46)	PFS89/0
10	Multiply Carry Miscompare byte 3	PFSALN(47)	PFS89/1
11	Multiply Carry Miscompare byte 4	PFSALN(48)	PFS89/2
12	Multiply Carry Miscompare byte 5	PFSALN(49)	PFS89/3
13	Multiply Carry Miscompare byte 6	PFSALN(50)	PFS89/4
14	Multiply Carry Miscompare byte 7	PFSALN(51)	PFS89/5
15	Multiply Carry Miscompare byte 8	PFSALN(52)	PFS89/6
16	Multiply C Register Parity Error byte 0	PFSALN(0)	PFS88/16
17	Multiply C Register Parity Error byte 1	PFSALN(1)	PFS88/17

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
18	Multiply C Register Parity Error byte 2	PFSALN(2)	PFS88/18
19	Multiply C Register Parity Error byte 3	PFSALN(3)	PFS88/19
20	Multiply C Register Parity Error byte 4	PFSALN(4)	PFS88/20
21	Multiply C Register Parity Error byte 5	PFSALN(5)	PFS88/21
22	Multiply C Register Parity Error byte 6	PFSALN(6)	PFS88/22
23	Multiply C Register Parity Error byte 7	PFSALN(7)	PFS88/23
24	Multiply B Register Parity Error byte 0	PFSALN(8)	PFS88/24
25	Multiply B Register Parity Error byte 1	PFSALN(9)	PFS88/25
26	Multiply B Register Parity Error byte 2	PFSALN(10)	PFS88/26
27	Multiply B Register Parity Error byte 3	PFSALN(11)	PFS88/27
28	Multiply B Register Parity Error byte 4	PFSALN(12)	PFS88/28
29	Multiply B Register Parity Error byte 5	PFSALN(13)	PFS88/29
30	Multiply B Register Parity Error byte 6	PFSALN(14)	PFS88/30
31	Multiply B Register Parity Error byte 7	PFSALN(15)	PFS88/31
32	DAI Data Parity Error byte 0	PFSOPI(27)	PFS84/48
33	DAI Data Parity Error byte 1	PFSOPI(28)	PFS84/49
34	DAI Data Parity Error byte 2	PFSOPI(29)	PFS84/50
35	DAI Data Parity Error byte 3	PFSOPI(30)	PFS84/51
36	DAI Data Parity Error byte 4	PFSOPI(31)	PFS84/52
37	DAI Data Parity Error byte 5	PFSOPI(32)	PFS84/53
38	DAI Data Parity Error byte 6	PFSOPI(33)	PFS84/54
39	DAI Data Parity Error byte 7	PFSOPI(34)	PFS84/55
40	ALN Micrand Parity Error byte 0	PFSALN(54)	PFS89/8
41	ALN Micrand Parity Error byte 1	PFSALN(55)	PFS89/9
42	ALN Micrand Parity Error byte 2	PFSALN(56)	PFS89/10
43	ALN Micrand Parity Error byte 3	PFSALN(57)	PFS89/11
44	ALN Micrand Parity Error byte 4	PFSALN(58)	PFS89/12
45	ALN Micrand Parity Error byte 5	PFSALN(59)	PFS89/13
46	ALN Micrand Parity Error byte 6	PFSALN(60)	PFS89/14

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
47	ALN Micrand Parity Error byte 7	PFSALN(61)	PFS89/15
48	Not Used		
49	Not Used		
50	Not Used		
51	Not Used		
52	Processor Did Not Halt Flag from PP Program		
53	Control Store Address bit 0		CSA/53
54	Control Store Address bit 1		CSA/54
55	Control Store Address bit 2		CSA/55
56	Control Store Address bit 3		CSA/56
57	Control Store Address bit 4		CSA/57
58	Control Store Address bit 5		CSA/58
59	Control Store Address bit 6		CSA/59
60	Control Store Address bit 7		CSA/60
61	Control Store Address bit 8		CSA/61
62	Control Store Address bit 9		CSA/62
63	Control Store Address bit 10		CSA/63

Section 1, Subsection 0

This subsection tests C register select of shifted data and sign extension from input bit 0.

Hardware Tested:

- C register for input of shifted data.
- Minor cycle control for C register (V0, select shifted data as input).
- Sign extension to upper 8 bits of C register.

Method:

The testing of the C register select of shifted data and sign extension from bit 0 is done under major and minor cycle control in the following manner.

The first micrand will transfer RFOC to ADATC, enable entering the C register from the ADATC bus, start the multiply/divide unit and cause the following minor cycle controls to execute from the ALN soft control memory:

- Cycle 1. Enable the C register and select the RDSC data input. Enable the C register into the Multiply/Divide Output mux.
- Cycle 2. Enable the C register and select the right-shifted data input. Enable the C register into the Multiply/Divide Output mux.
- Cycle 3. Enable the C register into the Multiply/Divide Output mux.

This makes C register bits 8 through 71 available to the ALN General network.

- Cycle 4. This soft control memory location does not have a load enable in it; therefore, the controls for cycle 3 will be re-executed.

The second micrand will transfer the Multiply/Divide Output mux through the ALN General network shifter logic and write it into RFOA. The ALN soft control locations executed by this micrand will not have the enter enable bit set. Therefore, the Multiply/Divide Output mux will contain the results of the previous micrand.

Results bits 0 through 3 will not be reported as they come from the Multiply Adder and are unknown at this time. Refer to the Section Method for status reporting.

Micrands Used:

Sequence Number	Description
5	Load C register with 64 bits and sign extend bit 0. (This sequence uses ALN soft control entry points 0 and 1.)

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RFOC.
a. Report test pattern; informational.

C.2 Test C register shift.
a. Start processor at Micrand Sequence 5; load C register with 64 bits and sign extend bit 0.
b. Wait halt.
c. Report RFOA, bits 4 through 63 contents.
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

Iteration Count	Test Pattern	Expected Result
0	0000 0000 0000 0000	0000 0000 0000 0000
1	FFFF FFFF FFFF FFFF	0FEF FFFF FFFF FFFF
2	5555 5555 5555 5555	0005 5555 5555 5555
3	AAAA AAAA AAAA AAAA	0FEA AAAA AAAA AAAA
4	1234 5678 1234 5678	0001 2345 6781 2345
5	8000 0000 0000 0000	0FE8 0000 0000 0000
6	7FFF FFFF FFFF FFFF	0007 FFFF FFFF FFFF

Section 1, Subsection 1

This subsection tests 32-bit sign extension in Divide Quotient register and C register entry from Quotient register.

Hardware Tested:

- C register entry controls for select of Divide Quotient register to C register bits 8 through 40.
- C register sign selection logic for select of input bit 32 to be extended.
- Sign extension using the Divide Quotient register.
- Minor cycle control bit V3; enable C register sign extension in Divide Quotient register.

Method:

The 32-bit sign selection, extension and entry into C register is tested in the following manner.

Two types of major cycle controls will be used to enable entry into the C register from the pattern contained in RFOC.

1. Sign extension of input bit 32 into C register bits 0 through 39 and input data in bits 40 through 71 (C register control = U6); Microrand Sequence 6.

2. Sign extension of input bit 32 into C register bits 0 through 7 and 24 through 39 and input data in C register bits 8 through 23 and 40 through 71 (C register control = U4, U6); Microrand Sequence 7.

Both sets of major cycle controls use the same ALN soft control. This gates the selected sign, extends it to the Divide Quotient register and enters it into the C register. The C register is then right-shifted and bits 8 through 71 are enabled to the Multiply/Divide Output mux.

The ALN general network transfers the Multiply/Divide Output mux through its shifter and its results are written into RFOA. This microrand does not alter the Multiply/Divide Output mux controls selected by the previous microrand.

Result bits 0 through 3 will not be reported as they come from the Multiply Full Adder and are unknown at this time.

Refer to Section Method for status reporting.

Isolation Notes:

- A 4-bit failure group in sign extension points to the failing divide array.
- An 8-bit failure group in sign extension points to the failing C register array if it falls on a C register entry boundary.
- Sign entered correctly into C register bits 0 through 7 (result bits 4 through 11) but no sign in C register bits 8 through 40 indicates a problem with C register entry controls or gated sign selection and the enable of sign extension by the V3 Minor Cycle Control register bit.
- A single bit failure points to the failing C register or divide array.

Microrands Used:

Both sequences use ALN soft control entry points 10(HEX) and 1.

Sequence Number	Description
6	Load C register with 32 bits, sign extended.
7	Load C register with input bits 0 through 15 and 32 through 63, sign extended from bit 32.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RFOC.
a. Report test pattern; informational.

Conditions:

C.2 Test sign extension from bit 32.
- a. Start processor at Microrand Sequence 6; load C register with 32 bits, sign extended.
- b. Wait halt.
- c. Report RF0A bits 4 through 63 contents.
- d. Error if not equal to expected result.

C.3 Read/format and report status.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Test sign extension from bit 32 or bit 0.
- a. Start processor at Microrand Sequence 7; load C register with input bits 9 through 15 and 32 through 63.
- b. Wait halt.
- c. Report RF0A bits 4 through 63 contents.
- d. Error if not equal to expected result.

C.5 Read/format and report status.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Report First Failure parity information.
- a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

Iteration Count	Test Pattern	Conditions	Expected Result
0	0000 0000 0000 0000	C.2	0000 0000 0000 0000
		C.4	0000 0000 0000 0000
1	0000 0000 8000 0000	C.2	0FFF FFFF FFF8 0000
		C.4	0FF0 000F FFF8 0000
2	FFFF FFFF 0000 0000	C.2	0000 0000 0000 0000
		C.4	0FFF FFFE 0000 0000
3	7FFF FFFF 8000 0000	C.2	0FFF FFFF FFF8 0000
		C.4	0FF7 FFFF FFF8 0000
4	7FFF FFFF 7FFF FFFF	C.2	0000 0000 0007 FFFF
		C.4	0007 FFF0 0007 FFFF

Section 1, Subsection 2

This subsection tests C sign selection logic under 170/180 mode microrand control and its entry into the C register.

Hardware Tested:

- C register entry controls for select of sign extension.
- C sign selection logic.
- U8 170 mode select.

Method:

The 170/180 mode flag in the ALN microrand controls the selection of input bit 0 or bit 1 as the sign to be extended into the C register bits 0 through 23 and is tested in the following manner.

The major cycle controls will enable one of two types of entry into the C register from the pattern contained in RF0C.

1. Sign extend input bit 0 into C register bits 0 through 23 (180 mode and C register controls = U4, U6).

2. Sign extend input bit 0 or bit 1 into C register bits 0 through 23 (170 mode and C register controls = U4, U6).

Both sets of major cycle controls use the same ALN soft control. This gates the selected sign, extends it to the Divide Quotient register and enters it in the enabled portion of the C register. The C register is then right-shifted and bits 8 through 71 are enabled to the Multiply/Divide Output mux.

The ALN General network transfers the Multiply/Divide Output mux through its Shifter network and the result is written into RF0A. This microrand's soft control does not alter the Multiply/Divide Output mux controls.

Results bits 0 through 3 are not reported as they come from the Multiply Final Adder and are unknown at this time.

Refer to Section Method for status reporting.

Microrands Used:

Both sequences use ALN soft control entry points 10(HEX) and 1.

Sequence Number	Description
8	Load C register with 48 bits, zero filled, 180 mode.
9	Load C register with 48 bits, sign extended bit 1, 170 mode.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0C.
a. Report test pattern; informational.

C.2 Test bit 0 sign extension.
a. Start processor at Micrand Sequence 8; load C with 48 bits, 180 mode.
b. Wait halt.
c. Report RF0A bits 4 through 63 contents.
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Test bit 0 or 1 sign extension.
a. Start processor at Micrand Sequence 9; load C with 48 bits, 170 mode.
b. Wait halt.
c. Report RF0A bits 4 through 63 contents.
d. Error if not equal to expected result.

C.5 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

Iteration Count	Test Pattern	Conditions	Expected Result
0	3FFF FFFF FFFF FFFF	C.2	0000 000F FFFF FFFF
		C.4	0000 000F FFFF FFFF
1	8000 0000 0000 0000	C.2	0000 0000 0000 0000
		C.4	0000 0000 0000 0000
2	4000 0000 0000 0000	C.2	0000 0000 0000 0000
		C.4	0FFF FFF0 0000 0000
3	0000 0000 8000 0000	C.2	0000 0000 0008 0000
		C.4	0000 0000 0008 0000

Section 1, Subsection 3

This subsection tests unsigned C register entry from major and minor cycle controls.

Hardware Tested:

- V8 control of C register bits 0 through 7 entry.
- U4 XOR U13 control of C register bits 8 through 23 entry.

Method:

Testing of the unsigned entry into the C register is done in the following manner.

The major cycle control will enable one of two types of entry into the C register.

1. U4 XOR U13 will sign extend input bit 0 into C register bits 8 through 23.
2. NOT U4 XOR U13 will cause ADATC bits 0 through 15 to be entered into C register bits 8 through 23.

Both sets of major cycle controls use the same ALN soft control. This gates the selected sign, extends it in the Divide Quotient register and enters it into the enabled portion of the C register. The entry sign into C register bit 0 through 7 is blocked by V8. The C register is right-shifted and bits 8 through 71 are enabled to the Multiply/Divide Output mux.

The Multiply/Divide Output mux is transferred through the ALN General network and written into RF0A.

Results bits 0 through 3 are not reported as they come from the Multiply Final Adder and are unknown at this time.

Refer to Section Method for status reporting.

Micrands Used:

Both sequences use ALN soft control entry points 11(HEX) and 1.

Sequence Number	Description
12	Load C register unsigned/signed.
13	Load C register with 64 bits unsigned.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0C.
a. Report test pattern; informational

C.2 Test C register entry with U4 XOR U13.
a. Start processor at Micrand Sequence 12; load C register unsigned/signed.
b. Wait halt.
c. Read and report RF0A bits 4 through 63 contents.
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Test C register entry with NOT U4 XOR U13.
a. Start processor at Micrand Sequence 13; load C register with 64 bits unsigned.
b. Wait halt.
c. Read and report RF0A bits 4 through 63 contents.
d. Error if not equal to expected result.

C.5 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

Iteration Count	Test Pattern	Conditions	Expected Result
0	5555 5555 5555 5555	C.2	0000 0005 5555 5555
		C.4	0005 5555 5555 5555
1	AAAA AAAA AAAA AAAA	C.2	000F FFFA AAAA AAAA
		C.4	000A AAAA AAAA AAAA
2	7FFF FFFF FFFF FFFF	C.2	0000 000F FFFF FFFF
		C.4	0007 FFFF FFFF FFFF
3	8000 0000 0000 0000	C.2	000F FFF0 0000 0000
		C.4	0008 0000 0000 0000

SECTION 2 - BASIC CHECKS OF MULTIPLY LOGIC

This section verifies operation of the following logic: B register, A and B Product networks, and the multiplier decoders. It also partially checks the following logic: Multiply Partial Summing network, Multiply Final Adder and Multiply/Divide Output mux.

Subsection 0 - Test that the multiply logic does not pick up bits when multiplying zero times zero and one times zero.

Subsection 1 - Test the following areas of the multiply logic: B register, B Product network for 1x decodes and partial sum generation, Multiply Partial Summing network for partial sum generation, the Final Adder and Multiply/Divide Output mux.

Subsection 2 - Test the following areas of the logic: A Product network for 1x decode and partial sum generation, and the Multiply Partial Summing network for partial sum generation.

Subsection 3 - Test B Product network for partial carry generation and the Multiply Partial Summing network for partial sum generation.

Subsection 4 - Test the A Product network for partial carry generation and the Multiply Partial Summing network for partial sum generation.

Subsection 5 - Test B Product network for partial carry generation from 1x decodes in C register bits 69 through 71 and 67 through 69.

Subsection 6 - Test A Product network for partial carry generation from 1x decodes in C register bits 63 through 65 and 61 through 63.

Subsection 7 - Test B Product network for partial carry generation from 1x decodes in C register bits 71 through 73 and 69 through 71.

Subsection 8 - Test A Product network for partial carry generation from 1x decodes in C register bits 65 through 67 and 63 through 65.

Subsection 9 - Test B Product network for partial carry and partial sum generation.

Subsection 10 - Test A Product network for partial carry and partial sum generation.

Subsection 11 - Test Multiply Partial Summing network for summing of an A Product network partial sum with a B Product network partial sum.

Subsection 12 - Test Multiply Partial Summing network for summing of an A Product network partial sum with a B Product network partial carry.

Subsection 13 - Test the Multiply Partial Summing network for the summing of a partial carry from the A Product with a partial sum from the B Product network.

Subsection 14 - Test the Multiply Partial Summing network for partial carry 2 generation and the Multiply Final Adder for adding of multiply partial carry bits.

Subsection 15 - Test the Multiply Partial Summing network for generation of N+2 carry.

Subsection 16 - Test the Multiply Partial Summing network for summing of a multiply partial sum bit with a Net B partial carry bit.

Subsection 17 - Test the Multiply Partial Summing network for summing of a multiply partial carry bit with a Net B partial sum bit.

Subsection 18 - Test 1x, -2x and Complement Correction decodes from C register bits 65 through 73 to the B and A Product networks.

Subsection 19 - Test 1x, -2x and Complement Correction decodes from C register bits 65 through 73 to the B and A Product networks and the Upper Multiply Partial Summing network.

Subsection 20 - Test 1x, -2x and Complement Correction decodes from C register bits 61 through 67 to the A Product network.

Subsection 21 - Test the 1x, -2x and Complement Correction decodes from C register bits 61 through 67 to the A Product network and the upper bits of the Multiply Partial Summing network.

Subsection 22 - Test 2x decodes to the B Product network.

Subsection 23 - Test the 2x decodes to the B Product network and the upper bits of the Multiply Partial Summing network.

Subsection 24 - Test 2x decodes to the A Product network.

Subsection 25 - Test the 2x decodes to the A Product network and the upper bits of the Multiply Partial Summing network.

Subsection 26 - Test -1x decodes to the B Product network.

Subsection 27 - Test the -1x decodes to the B Product network and the upper bits of the Multiply Partial Summing network.

Subsection 28 - Test -1x decodes to the A Product network.

Subsection 29 - Test the -1x decodes to the A Product network and the upper bits of the Multiply Partial Summing network.

Subsection 30 - Test -1x decodes to the B Product network.

Subsection 31 - Test the -1x decodes to the B Product network and the upper bits of the Multiply Partial Summing network.

Subsection 32 - Test -1x decodes to the A Product network.

Subsection 33 - Test the -1x decodes to the A Product network and the upper bits of the Multiply Partial Summing network.

Subsection 34 - Test -0x decodes to the B Product network.

Subsection 35 - Test the -0x decodes to the B Product network and the upper bits of the Multiply Partial Summing network.

Subsection 36 - Test the -0x decodes to the A Product network.

Subsection 37 - Test the -0x decodes to the A Product network and the upper bits of the Multiply Partial Summing network.

Subsection 38 - Test C register sign control of complement decodes with decodes in C register bits 67 through 73.

0 Subsection 39 - Test C register sign control of complement decodes with decodes in C register bits 61 through 67.

Method:

General:

All testing in this section makes use of the hardware design that does a 12-bit by 64-bit multiply each minor cycle and that provides control of each minor cycle by the ALN soft control memory. These controls are used to do one of two types of multiplies: a straight 12-by-64 bit multiply with the results taken directly as a true result from the Final Adder or as a 24-by-64 bit, 2-iteration, 12-by-64, 12-by-64 multiply with the partial sums and partial carries of the first iteration summed with the partial sums and partial carries of the second iteration and with the result taken from the Final Adder.

Hardware Controls:

Major Cycle Control: Both micrand sequences used in this section use a similar procedure and only vary in the minor cycle control that they cause to be executed in the Multiply/Divide unit. The first micrand loads the B register with 64 bits from RF0B and the C register with 64 bits from RF0C. The sequence copies the Multiply/Divide Output mux through the ALN General network and writes it into RF0A. The sequence then copies the Multiply/Divide Output mux through the ALN General network and writes it in RF09. The basic difference between the two micrand sequences used is that an extra micrand is needed when enabling the partial summary hardware at an earlier minor cycle time to allow for propper data latching.

Minor cycle control: Two types of minor cycle control are used in this section.

a. Nonfeedback multiply (ALN soft control entry points 0, 1 and 2) causes the following sequence of operations in the multiply logic: 1. The C register is enabled but the corresponding enable for clocking of the Multiply Partial Summing network is not set. This is done to overcome the fact that the C register adds 2 bits below the least significant bits of the multiplier and skews the multiplier decodes. The bit not being set effectively discards the results of the multiply by the lower 10 bits of the multiplier. 2. Parity checking is enabled, the C register shifted and the clocking of the Multiply Partial Summing network enabled. 3. The Multiply/Divide Output mux is switched to select MSUM bits 8 through 71 and the minor cycle controls are held in this condition so that the second micrand can read this value. 4. When the third micrand is executed the minor cycle controls are changed to cause the Multiply/Divide Output mux to select the MSUM bits 0 through 7.

b. Feedback multiply (ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX)) causes the following sequence of operations in the multiply logic: 1. The C register is enabled but the corresponding enable for clocking of the Multiply Partial Summing network is not set. 2. Parity checking is enabled, the C Register is shifted, the checking is enabled, the C register shifted and the clocking of the Multiply Partial Summing network enabled, and the Multiply Partial Summing network feedback is also enabled. 3. The Multiply/Divide Output mux is switched to select the MSUM bits 8 through 71 and the minor cycle controls are held in this condition so that the third micrand can read this value. 4. When the fourth micrand is executed, the minor cycle controls are changed to cause the Multiply/Divide Output mux to select the MSUM bits 0 through 7.

Status Reporting:

All conditions in this section which read and report status will read and check all statuses listed in the section status format.

Micrands Used:

Sequence Number	Description
10	Run multiply from ALN soft control entry points 0, 1, and 2. This sequence loads the Multiply B register from RF0B, loads the C register from RF0C, and writes the Multiply/Divide Output mux in RF0A and RF09.
67	Run multiply from ALN soft control entry points 10, 11, 12, and 13(HEX). This sequence loads the Multiply B register from RF0B, loads the C register from RF0C, and writes the Multiply/Divide Output mux in RF0A and RF09.

Initialization:

- Done as part of subsection 0 PP code.
- Write the following ALN soft control memory map into the locations specified.

Location	Value to be Written
0	4108 4108
1	D308 0000
2	0108 0000
3	0000 0000
4	0114 D308
5	0000 D908
6	0000 0108
7	0000 0000
8	010C 0114
9	0000 0000
A	0000 0000
B	0000 0000
C	0000 010C
D	0000 0000
E	0000 0000
F	0000 0000

Section Status Format:

The following bit assignments are used by all conditions that report status.

NOTE

Any status not read by a condition and all unused bit positions will be reported as zero.

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
0	Soft Control Register Parity Error byte 0	PFSALN(62)	PFS88/14
1	Soft Control Register Parity Error byte 1	PFSALN(63)	PFS88/15
2	Not Used		
3	Not Used		
4	Not Used		
5	Not Used		
6	Multiply Carry Miscompare byte 0	PFSALN(44)	PFS88/60
7	Multiply Carry Miscompare byte 1	PFSALN(45)	PFS88/61
8	Not Used		
9	Multiply Carry Miscompare byte 2	PFSALN(46)	PFS89/0
10	Multiply Carry Miscompare byte 3	PFSALN(47)	PFS89/1
11	Multiply Carry Miscompare byte 4	PFSALN(48)	PFS89/2
12	Multiply Carry Miscompare byte 5	PFSALN(49)	PFS89/3
13	Multiply Carry Miscompare byte 6	PFSALN(50)	PFS89/4
14	Multiply Carry Miscompare byte 7	PFSALN(51)	PFS89/5
15	Multiply Carry Miscompare byte 8	PFSALN(52)	PFS89/6
16	Multiply C Register Parity Error byte 0	PFSALN(0)	PFS88/16
17	Multiply C Register Parity Error byte 1	PFSALN(1)	PFS88/17
18	Multiply C Register Parity Error byte 2	PFSALN(2)	PFS88/18
19	Multiply C Register Parity Error byte 3	PFSALN(3)	PFS88/19
20	Multiply C Register Parity Error byte 4	PFSALN(4)	PFS88/20
21	Multiply C Register Parity Error byte 5	PFSALN(5)	PFS88/21
22	Multiply C Register Parity Error byte 6	PFSALN(6)	PFS88/22
23	Multiply C Register Parity Error byte 7	PFSALN(7)	PFS88/23
24	Multiply B Register Parity Error byte 0	PFSALN(8)	PFS88/24
25	Multiply B Register Parity Error byte 1	PFSALN(9)	PFS88/25
26	Multiply B Register Parity Error byte 2	PFSALN(10)	PFS88/26
27	Multiply B Register Parity Error byte 3	PFSALN(11)	PFS88/27

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
28	Multiply B Register Parity Error byte 4	PFSALN(12)	PFS88/28
29	Multiply B Register Parity Error byte 5	PFSALN(13)	PFS88/29
30	Multiply B Register Parity Error byte 6	PFSALN(14)	PFS88/30
31	Multiply B Register Parity Error byte 7	PFSALN(15)	PFS88/31
32	DAI Data Parity Error byte 0	PFSOPI(27)	PFS84/48
33	DAI Data Parity Error byte 1	PFSOPI(28)	PFS84/49
34	DAI Data Parity Error byte 2	PFSOPI(29)	PFS84/50
35	DAI Data Parity Error byte 3	PFSOPI(30)	PFS84/51
36	DAI Data Parity Error byte 4	PFSOPI(31)	PFS84/52
37	DAI Data Parity Error byte 5	PFSOPI(32)	PFS84/53
38	DAI Data Parity Error byte 6	PFSOPI(33)	PFS84/54
39	DAI Data Parity Error byte 7	PFSOPI(34)	PFS84/55
40	ALN Micrand Parity Error byte 0	PFSALN(54)	PFS89/8
41	ALN Micrand Parity Error byte 1	PFSALN(55)	PFS89/9
42	ALN Micrand Parity Error byte 2	PFSALN(56)	PFS89/10
43	ALN Micrand Parity Error byte 3	PFSALN(57)	PFS89/11
44	ALN Micrand Parity Error byte 4	PFSALN(58)	PFS89/12
45	ALN Micrand Parity Error byte 5	PFSALN(59)	PFS89/13
46	ALN Micrand Parity Error byte 6	PFSALN(60)	PFS89/14
47	ALN Micrand Parity Error byte 7	PFSALN(61)	PFS89/15
48	Not Used		
49	Not Used		
50	Not Used		
51	Not Used		
52	Processor Did Not Halt Flag from PP Program		
53	Control Store Address bit 0		CSA/53
54	Control Store Address bit 1		CSA/54
55	Control Store Address bit 2		CSA/55
56	Control Store Address bit 3		CSA/56
57	Control Store Address bit 4		CSA/57
58	Control Store Address bit 5		CSA/58
59	Control Store Address bit 6		CSA/59
60	Control Store Address bit 7		CSA/60
61	Control Store Address bit 8		CSA/61
62	Control Store Address bit 9		CSA/62
63	Control Store Address bit 10		CSA/63

Section 2, Subsection 0

This subsection tests that the Multiply logic does not pick up bits when multiplying zero times zero and one times zero.

Hardware Tested:

The following areas of the logic are checked to ensure that no bits are being picked.

- B register.
- A Product network.
- B Product network.
- Multiply Partial Summing network.
- Multiply Full Adder.
- Multiply/Divide Output mux for select of Adder input.

Method:

The complete Multiply unit logic is checked for picking of bits by multiplying zero times zero and one times zero using the hardware controls described in the Section Method.

Isolation Notes:

Both multipliers will be run twice: the first time without feedback enabled in the Multiply Partial Summing network and the second time with feedback enabled. This is done to provide isolation between the logic before the Multiply Partial Summing network and after it. The two different multipliers are used to provide isolation in the product network and the B register. Below are categories of failures and the areas of the logic which they isolate to:

- Picking the same bit on all multiplies indicates a problem in the Output mux, the Adder, or the Partial Summing network for that bit.

- Picking 1 bit on the nonfeedback multiply and 2 bits separated by 12 bit positions in the feedback multiply indicates a problem in the Partial Summing network or the B Product network. Note, however, that this is not valid when the nonfeedback multiply picks a bit in the lower 12 bits.

- If a zero times zero multiply picks bits but one times zero does not, it indicates a problem in the B Product network.

- If a zero times one picks bits but zero times zero does not, it indicates a problem in the B Product network or the B register.

- Carry error isolates to the adder array providing that status, the next least significant adder array, or the carry interchange.

- B or C register parity errors isolate to the array providing the error, the Micrand Aux, or the Processor Fault Status registers.

Conditions:

C.0 Run zero times zero without feedback.
a. Write 0000 0000 0000 0000 in RF0B.
b. Write 0000 0000 0000 0000 in RF0C.
b. Start processor at Micrand Sequence 10; run multiply from ALN soft control address 0, 1, and 2.
d. Wait halt.
e. Read and report contents of RF0A (MSUM bits 8 through 71).
f. Error if not equal to 0000 0000 0000 0000(HEX).

C.1 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to 0000 0000 0000 0000(HEX).

C.2 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.3 Run zero times zero with feedback enabled.
a. Start processor at Micrand Sequence 67; run multiply from ALN soft control address 10, 11, 12, and 13.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to 0000 0000 0000 0000.

C.4 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to 0000 0000 0000 0000.

C.5 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Run one times zero without feedback.
a. Write 0000 0000 0000 080016 in RF0C.
b. Start processor at Micrand Sequence 10; run multiply from ALN soft control address 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to 0000 0000 0000 0000(HEX).

C.7 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to 0000 0000 0000 0000(HEX).

C.8 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

Conditions:

C.9 Run one times zero with feedback enabled.
- a. Start processor at Microrand Sequence 67; run multiply from ALN soft control address 10, 11, 12, and 13.
- b. Wait halt.
- c. Read and report contents of RF0A (MSUM bits 8 through 71).
- d. Error if not equal to 0000 0000 0000 0000.

C.10 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
- a. Error if not equal to 0000 0000 0000 0000.

C.11 Read/format and report status.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.12 Report First Failure parity information.
- a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

Section 2, Subsection 1

This subsection tests the following areas of the logic: B register, B Product network for 1x decode and partial sum generation, and the partial sum path through the Multiply Partial Summing network and Multiply Final Adder.

Hardware Tested:

- B register data path.
- B Product network 1x decodes of C register bits 6 through 73.
- B Product network partial sum generation.
- Multiply Partial Summing network for passing of B Product network partial sums to its output and looping them back when feedback is enabled.
- Multiply Full and Iteration Adder for adding of multiply partial sums to zeros.
- Multiply/Divide Output mux for selection of MSUM bits from Adders.

Method:

The testing of the B register, B Product network decodes and Partial Sums, Multiply Partial Sum register, Multiply Final Adder and Output mux is done using the hardware controls described in the Section Method and the multipliers selected to pull the three 1x decodes into the B Product network. Each test pattern will be run against each of the decodes without feedback enabled. Then the third decode will be repeated with feedback enabled.

Each multiply result will be reported as the output of the Multiply Final Adders (MSUM bits 0 through 71) and all status defined in the section status format will also be reported.

Isolation Notes:

The following are descriptions of general error classes that isolate to specific areas of the Multiply logic.

- A bit or multibit failures that stay in the same output bits for all multipliers, indicate a problem in the Final Adder or Output mux.

- A bit or multibit failures that stay the same for all nonfeedback multiplies but show up in two places in the feedback multiply indicate a problem in the Multiply Partial Summing network.

NOTE

Not valid for lower 12 bits or the upper 24 bits of the Multiply Partial Summing network.

- A bit or multibit failures that stay the same in nonfeedback multiply and move right 12 places in feedback multiply indicate a problem in the B Product network output or the Multiply Partial Summing network.

- A bit or multibit failures in only one nonfeedback condition indicate a problem in the B decode or Product network.

- A bit or multibit failures in all nonfeedback condition with the failure left-shifted by two between conditions is in the B Product network or the B register.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.

C.2 Multiply test pattern by 1x decode from C register bits 71 through 73 without feedback.
- a. Write multiplier of 0000 0000 0000 0800(HEX) in RF0C.
- b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
- c. Wait halt.
- d. Read and report contents of RF0A (MSUM bits 8 through 71).
- e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
- a. Error if not equal to expected result.

Conditions:

C.4 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Multiply test pattern by 1x decode from C register bits 69 through 71 without feedback.
a. Write multiplier of 0000 0000 0000 2000(HEX) in RF0C.
b. Start processor at Micrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Multiply test pattern by 1x decode from C register bits 67 through 69 without feedback.
a. Write multiplier of 0000 0000 0000 8000(HEX) in RF0C.
b. Start processor at Micrand Sequence 10; run multiply using ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Multiply test pattern by 1x decode from C register bits 67 through 69 with feedback.
a. Start processor at Micrand Sequence 67; run multiply using ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.12 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.13 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.14 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Conditions 2, 5, 8, and 11 report bits 8 through 71 as bits 0 through 63.
Conditions 3, 6, 9, and 12 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
		C.11 and C.12	00	0000 0000 0000 0000
1	7FFF FFFF FFFF FFFF	C.2 and C.3	00	7FFF FFFF FFFF FFFF
		C.5 and C.6	01	FFFF FFFF FFFF FFFC
		C.8 and C.9	07	FFFF FFFF FFFF FFF0
		C.11 and C.12	00	007F FFFF FFFF FFFF
2	2AAA AAAA AAAA AAAA	C.2 and C.3	00	2AAA AAAA AAAA AAAA
		C.5 and C.6	00	AAAA AAAA AAAA AAA8
		C.8 and C.9	02	AAAA AAAA AAAA AAA0
		C.11 and C.12	00	002A AAAA AAAA AAAA
3	5555 5555 5555 5555	C.2 and C.3	00	5555 5555 5555 5555
		C.5 and C.6	01	5555 5555 5555 5554
		C.8 and C.9	05	5555 5555 5555 5550
		C.11 and C.12	00	0055 5555 5555 5555
4	7CDC DCDC DCDC DCDC	C.2 and C.3	00	7CDC DCDC DCDC DCDC
		C.5 and C.6	01	F373 7373 7373 7370
		C.8 and C.9	07	CDCD CDCD CDCD CDC0
		C.11 and C.12	00	007C DCDC DCDC DCDC

Section 2, Subsection 2

This subsection tests the following areas of the logic: A Product network for 1x decode and partial sum generation and its path through the Multiply Partial Summing network.

Hardware Tested:

- A Product network 1x decode of C register bits 61 through 67.
- A Product network for partial sum generation.
- Multiply Partial Summing network for inputs from A Product network.

Method:

Testing of the A Product network decodes and Multiply Partial Summing is performed using the hardware controls described in the Section Method and the multipliers selected to pull the three 1x decodes in the A Product network. Each of the test patterns will be run against each of the decodes without feedback enabled. Then the third decode will be rerun with feedback enabled.

Each multiply result will be reported as the output of Multiply Final Adder (MSUM bits 0 through 71) and all status defined in the Section Status Format will be reported.

Isolation Notes:

- Except for MSUM bits 0 through 4, the output path through the Adder and Multiply/Divide mux has been tested in subsection 1 and, therefore, all errors in MSUM bits 5 through 71 isolate to the A Product network or the Multiply Partial Summing network for the nonfeedback multiplies.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Multiply test pattern by 1x decode from C register bits 65 through 67 without feedback.
a. Write multiplier 0000 0000 0002 0000(HEX) in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

Conditions:

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Multiply test pattern by 1x decode from C register bits 63 through 65 without feedback.
a. Write multiplier of 0000 0000 0008 0000(HEX) in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Multiply test pattern by 1x decode from C register bits 61 through 63 without feedback.
a. Write multiplier of 0000 0000 0020 0000(HEX) in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Multiply test pattern by 1x decode from C register bits 61 through 63 with feedback enabled.
a. Start processor at Microrand Sequence 67; run multiply from soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.12 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.13 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.14 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Conditions 2, 5, 8, and 11 report bits 8 through 71 as bits 0 through 63. Conditions 3, 6, 9, and 12 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
		C.11 and C.12	00	0000 0000 0000 0000
1	7FFF FFFF FFFF FFFF	C.2 and C.3	1F	FFFF FFFF FFFF FFC0
		C.5 and C.6	7F	FFFF FFFF FFFF FF00
		C.8 and C.9	FF	FFFF FFFF FFFF FC00
		C.11 and C.12	00	1FFF FFFF FFFF FFFF
2	2AAA AAAA AAAA AAAA	C.2 and C.3	0A	AAAA AAAA AAAA AA80
		C.5 and C.6	2A	AAAA AAAA AAAA AA00
		C.8 and C.9	AA	AAAA AAAA AAAA A800
		C.11 and C.12	00	0AAA AAAA AAAA AAAA
3	5555 5555 5555 5555	C.2 and C.3	15	5555 5555 5555 5540
		C.5 and C.6	55	5555 5555 5555 5500
		C.8 and C.9	55	5555 5555 5555 5400
		C.11 and C.12	00	1555 5555 5555 5555
4	7CDC DCDC DCDC DCDC	C.2 and C.3	1F	3737 3737 3737 3700
		C.5 and C.6	7C	DCDC DCDC DCDC DC00
		C.8 and C.9	F3	7373 7373 7373 7000
		C.11 and C.12	00	1F37 3737 3737 3737

Section 2, Subsection 3

This subsection tests B Product network partial carry generation and Multiply Partial Summing network for handling of the partial carries.

Hardware Tested:

- B Product network for MPC1B generation.
- Multiply Partial Summing network handling of MPC1B inputs.

Method:

The testing of the B Product network partial carry generation and its summing in the Multiply Partial Summing network is performed using the hardware controls described in the Section Method.

The multiplier for this subsection, 0000 0000 0000 8800(HEX), will pull the 1x decodes from C register bits 71 through 73 and 67 through 69. The multiplicand (test pattern) is structured such that the partial summing caused by the two 1x decodes causes only one partial carry to be generated. The Multiply Partial Summing network will sum this with zeros, and the result is reflected in the next higher PSR2 bit position. The test patterns start with the carry generated to MPC1B bit 67 and walk to the carry generated to MPC1B bit 9. The center bit of the 3 bits set in the result reflects the carry generation.

Initialization:

- Write 0000 0000 0000 8800(HEX) in RF0C.

Conditions:

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test MPC1B generation using nonfeedback multiply.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report RF0A contents (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

Conditions:

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 as bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0011	00	0000 0000 0000 0121
1	0000 0000 0000 0022	00	0000 0000 0000 0242
2	0000 0000 0000 0044	00	0000 0000 0000 0484
3	0000 0000 0000 0088	00	0000 0000 0000 0908
.	.		.
.	.		.
.	.		.
.	.		.
37	0880 0000 0000 0000	00	9080 0000 0000 0000
38	1100 0000 0000 0000	01	2100 0000 0000 0000
39	2200 0000 0000 0000	02	4200 0000 0000 0000
3A	4400 0000 0000 0000	04	8400 0000 0000 0000

Section 2, Subsection 4

This subsection tests A Product network for partial carry generation and Multiply Partial Summing network for handling of the partial carries.

Hardware Tested:

- A Product network for MPC1A generation.
- Multiply Partial Summing network for handling of MPC1A inputs.

Method:

The testing of the A Product network partial carry generation and its summing in the Multiply Partial Summing network is alone in using the hardware controls described in the Section Method.

The multiplier for this subsection, 0000 0000 0022 0000(HEX), will pull the 1x decodes from C register bits 65 through 67 and 61 through 63. The multiplicand test pattern is structured such that the partial summing caused by the two 1x decodes causes only one partial carry to be generated. The Multiply Partial Summing network will sum this with zeros and the result is reflected in the next higher PSR2 bit position. The test patterns start with the carry generated to the MPC1A bit 65 and walk to the carry generated to MPC1A bit 7. The center bit of the 3 bits set in the result reflects the carry generation.

Initialization:

- Write 0000 0000 0022 0000(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test MPC1A generation using nonfeedback multiply.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

Conditions:

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 as bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0011	00	0000 0000 0000 4840
1	0000 0000 0000 0022	00	0000 0000 0000 9080
2	0000 0000 0000 0044	00	0000 0000 0001 2100
3	0000 0000 0000 0088	00	0000 0000 0002 4200
.	.		.
.	.		.
.	.		.
.	.		.
37	0880 0000 0000 0000	24	2000 0000 0000 0000
38	1100 0000 0000 0000	48	4000 0000 0000 0000
39	2200 0000 0000 0000	90	8000 0000 0000 0000
3A	4400 0000 0000 0000	21	0000 0000 0000 0000

NOTE

When pattern 3A is run the result from the Multiply Partial Summing network is too large for the output path and the top bit is not output.

Section 2, Subsection 5

This subsection tests B Product network for partial carry generation from 1x decodes in C register bits 69 through 71 and 67 through 69.

Hardware Tested:

- B Product network.
- Multiply Partial Summing network for handling of MPC1B bits 7 and 8.

Method:

The testing of the B Product network for partial carry generation from 1x decodes is done using the hardware controls described in the Section Method.

The multiplier for this subsection, 0000 0000 0000 A000(HEX), pulls the 1x decodes from C register bits 69 through 71 and 67 through 69. The two decodes and the multiplicand (test pattern) should cause a specific carry to be generated. The test patterns start with MPC1B bit 67 and walk through MPC1B bit 7. The center bit of the 3 bits set in the result reflects the carry generation.

Initialization:

- Write 0000 0000 0000 A000(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

Conditions:

C.2 Test MPC1B generation using nonfeedback multiply.
- a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
- b. Wait halt.
- c. Read and report contents of RF0A (MSUM bits 8 through 71).
- d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
- a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
- a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 as bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0005	00	0000 0000 0000 0064
1	0000 0000 0000 000A	00	0000 0000 0000 00C8
2	0000 0000 0000 0014	00	0000 0000 0000 0190
3	0000 0000 0000 0028	00	0000 0000 0000 0320
.	.		.
.	.		.
.	.		.
.	.		.
39	0A00 0000 0000 0000	00	C800 0000 0000 0000
3A	1400 0000 0000 0000	01	9000 0000 0000 0000
3B	2800 0000 0000 0000	03	2000 0000 0000 0000
3C	5000 0000 0000 0000	06	4000 0000 0000 0000

Section 2, Subsection 6

This subsection tests A Product network for partial carry generation from 1x decodes in C register bits 63 through 65 and 61 through 63.

Hardware Tested:

- A Product network.
- Multiply Partial Summing network for handling of MPC1A bits 5 and 6.

Method:

The testing of the A Product network for partial carry generation from 1x decodes is done using the hardware controls described in the Section Method.

The multiplier for this subsection, 0000 0000 0028 0000(HEX), pulls the 1x decodes from C register bits 63 through 65 and 61 through 63. The two decodes along with the multiplicand (test pattern) should cause a specific carry to be generated. The test patterns start with MPC1A bit 65 and walk to MPC1A bit 5. The center bit of the 3 bits set in the result reflects the carry generation.

Initialization:

- Write 0000 0000 0028 0000(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
- a. Report test pattern; informational.

C.2 Test MPC1A generation using nonfeedback multiply.
- a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
- b. Wait halt.
- c. Read and report contents of RF0A (MSUM bits 8 through 71).
- d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
- a. Error if not equal to expected result.

Conditions:

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 as bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0005	00	0000 0000 0000 1900
1	0000 0000 0000 000A	00	0000 0000 0000 3200
2	0000 0000 0000 0014	00	0000 0000 0000 6400
3	0000 0000 0000 0028	00	0000 0000 0000 C800
.	.		.
.	.		.
.	.		.
39	0A00 0000 0000 0000	32	0000 0000 0000 0000
3A	1400 0000 0000 0000	64	0000 0000 0000 0000
3B	2800 0000 0000 0000	C8	0000 0000 0000 0000
3C	5000 0000 0000 0000	90	0000 0000 0000 0000

NOTE

When pattern 3C is run, the result from the Multiply Partial Summing network is too large for the output path and the top bit is not output.

Section 2, Subsection 7

This subsection tests B Product network for partial carry generation from 1x decodes in C register bits 71 through 73 and 69 through 71.

Hardware Tested:

- B Product network.
- Multiply Partial Summing network handling of MPC1B bits 68 and 69.

Method:

The testing of the B Product network for partial carry generation from 1x decodes is done using the hardware controls described in the Section Method.

The multiplier for this subsection, 0000 0000 0000 2800(HEX), pulls the 1x decodes from C register bits 71 through 73 and 69 through 71. The two decodes along with the multiplicand (test pattern) should cause a specific carry to be generated. The center bit of the 3 bits set in the result reflects the carry generation. The test patterns start with MPC1B bit 69 and walk to MPC1B bit 9.

Initialization:

- Write 0000 0000 0000 2800(HEX) in RF0C.

Conditions:

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test MPC1B generation using nonfeedback multiply.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

Conditions:

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to the Section Status Format.
b. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 as bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0005	00	0000 0000 0000 0019
1	0000 0000 0000 000A	00	0000 0000 0000 0032
2	0000 0000 0000 0014	00	0000 0000 0000 0064
3	0000 0000 0000 0028	00	0000 0000 0000 00C8
.	.		.
.	.		.
.	.		.
.	.		.
39	0A00 0000 0000 0000	00	3200 0000 0000 0000
3A	1400 0000 0000 0000	00	6400 0000 0000 0000
3B	2800 0000 0000 0000	00	C800 0000 0000 0000
3C	5000 0000 0000 0000	01	9000 0000 0000 0000

Section 2, Subsection 8

This subsection tests the A Product network for partial carry generation from 1x decodes in C register bits 65 through 67 and 63 through 65.

Hardware Tested:

- A Product network.
- Multiply Partial Summing network for handling of MPC1A bits 66 and 67.

Method:

Testing of the A Product network for partial carry generation from 1x decodes is done using the hardware controls described in the Section Method.

The multiplier for this subsection, 0000 0000 000A 0000(HEX), pulls the 1x decodes from C register bits 65 through 67 and 63 through 65. The two decodes along with the multiplicand (test pattern) should cause a specific carry to be generated. The test patterns start with MPC1A bit 65 and walk to MPC1A bit 5. The center bit of the 3 bits set in the result reflects the carry generation. The test patterns start with MPC1A bit 67 and walk through MPC1A bit 7.

Initialization:

- Write 0000 0000 000A 0000(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test MPC1A generation using nonfeedback multiply.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

Conditions:

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 as bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0005	00	0000 0000 0000 0640
1	0000 0000 0000 000A	00	0000 0000 0000 0C80
2	0000 0000 0000 0014	00	0000 0000 0000 1900
3	0000 0000 0000 0028	00	0000 0000 0000 3200
.	.		.
.	.		.
.	.		.
.	.		.
39	0A00 0000 0000 0000	0C	8000 0000 0000 0000
3A	1400 0000 0000 0000	19	0000 0000 0000 0000
3B	2800 0000 0000 0000	32	0000 0000 0000 0000
3C	5000 0000 0000 0000	64	0000 0000 0000 0000

Section 2, Subsection 9

This subsection tests B Product network for partial carry and partial sum generation from 1x decodes in all three C register bit groups.

Hardware Tested:

- B Product network.

Method:

Testing of the B Product network partial carry and sum generations from three 1x decodes is done using the hardware controls described in the Section Method.

The multiplier for this subsection, 0000 0000 0000 A800(HEX), will pull all three 1x decodes into the B Product network. These three decodes and the multiplicand (test pattern) should cause a partial sum and a partial carry to be generated in the same bit position. The test patterns start with MPS1B and MPC1B bit 67 and walk through MPS1B and MPC1B bit 9.

Initialization:

- Write 0000 0000 0000 A800(HEX) in RF0C.

Conditions:

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test MPC1B and MPS1B generation using nonfeedback multiply.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2).
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to the Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 in bits 0 through 63. Condition 3 reports expected bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0015	00	0000 0000 0000 01B9
1	0000 0000 0000 002A	00	0000 0000 0000 0372
2	0000 0000 0000 0054	00	0000 0000 0000 06E4
3	0000 0000 0000 00A8	00	0000 0000 0000 0DC8
.	.		.
.	.		.
.	.		.
.	.		.
37	0A80 0000 0000 0000	00	DC80 0000 0000 0000
38	1500 0000 0000 0000	01	B900 0000 0000 0000
39	2A00 0000 0000 0000	03	7200 0000 0000 0000
3A	5400 0000 0000 0000	06	E400 0000 0000 0000

Section 2, Subsection 10

This subsection tests A Product network for partial carry and partial sum generation from 1x decodes in all three C register bit groups.

Hardware Tested:

- A Product network.

Method:

Testing of the A Product network partial carry and sum generation from three 1x decodes is done using the hardware controls described in the Section Method.

The multiplier for this subsection, 0000 0000 002A 0000(HEX), will pull all three 1x decodes into the A Product network. These three decodes and the multiplicand (test pattern) should cause a partial sum and a partial carry to be generated in the same bit position. The test patterns start with MPS1A and MPC1A bit 65 and walk to MPS1A and MPC1A bit 7.

Initialization:

- Write 0000 0000 002A 0000(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
  a. Report test pattern; informational.

C.2 Test MPC1A and MPS1A generation using nonfeedback multiply.
  a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
  b. Wait halt.
  c. Read and report contents of RF0A (MSUM bits 8 through 71).
  d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
  a. Error if not equal to expected result.

C.4 Read/format and report status according to the Section Status Format.
  a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
  a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 in bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0015	00	0000 0000 0000 6E40
1	0000 0000 0000 002A	00	0000 0000 0000 DC80
2	0000 0000 0000 0054	00	0000 0000 0001 B900
3	0000 0000 0000 00A8	00	0000 0000 0003 7200
.	.		.
.	.		.
.	.		.
37	0A80 0000 0000 0000	37	2000 0000 0000 0000
38	1500 0000 0000 0000	6E	4000 0000 0000 0000
39	2A00 0000 0000 0000	DC	8000 0000 0000 0000
3A	5400 0000 0000 0000	B9	0000 0000 0000 0000

NOTE

When pattern 3A is run, the result from the Multiply Partial Summing network is too large for the output path and the top bit is not output.

Section 2, Subsection 11

This subsection tests Multiply Partial Summing network for summing a partial sum from the A Product network with a partial sum from the B Product network.

Hardware Tested:

- Multiply Partial Summing network partial summing and the carry out 1 and carry in 1 operation.

Method:

Testing of the Multiply Partial Summing network for summing of partial sums from the A and B Product network is done using the hardware controls described in the Section Method.

The multiplier for this subsection, 0000 0000 0002 0800(HEX), will pull the 1x decode in C register bits 65 through 67 to the A Product network and the 1x decode in C register bits 71 through 73 to the B Product network. These two decodes and the multiplicand (test pattern) should cause only one partial sum from the A Product network to be summed with a partial sum from the B Product network. The result of this summing is reflected in the next higher PSR2 bit position. When the result of the summing has to pass up to the next array it is done via the CROT (1) to CRIN (1) interarray connection.

The test patterns start with the summing of MPS1A bit 69 with MPS1B bit 65 and walk to the summing of MPS1A bit 13 with MPS1B bit 9.

Initialization:

- Write 0000 0000 0002 0800(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test MPS1A and MPS1B summing using nonfeedback multiply.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to the Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 expected bits 8 through 71 in bits 0 through 63. Condition 3 expected bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0041	00	0000 0000 0000 1081
1	0000 0000 0000 0082	00	0000 0000 0000 2102
2	0000 0000 0000 0104	00	0000 0000 0000 4204
3	0000 0000 0000 0208	00	0000 0000 0000 8408
.	.		.
.	.		.
.	.		.
.	.		.
35	0820 0000 0000 0000	02	1020 0000 0000 0000
36	1040 0000 0000 0000	04	2040 0000 0000 0000
37	2080 0000 0000 0000	08	4080 0000 0000 0000
38	4100 0000 0000 0000	10	8100 0000 0000 0000

Section 2, Subsection 12

This subsection tests Multiply Partial Summing network for summing a partial sum from the A Product network with a partial carry from the B Product network.

Hardware Tested:

- Multiply Partial Summing network.

Method:

Testing of the Multiply Partial Summing network for summing a partial sum from the A Product network with a partial carry from the B Product network is done using the hardware controls described in the Section Method.

The multiplier for this subsection, 0000 0000 0008 8800(HEX), pulls the 1x decode from C register bits 63 through 65 to the A Product network and the 1x decodes from C register bits 71 through 73 and 67 through 69. These decodes and the multiplicand (test pattern) should cause a partial carry from the B Product network to be summed with partial sum from the A Product network.

The test patterns start with the summing of MPS1A bit 67 with MPC1B bit 64 and walk through the summing of MPS1A bit 12 with MPC1B bit 9. The last four test patterns also test the summing of MPS1A bits 9 through 12 with MPS1B bits 5 through 8.

Initialization:

- Write 0000 0000 0008 8800(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test MPS1A and MPC1B summing using nonfeedback multiply.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to the Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 in bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0089	00	0000 0000 0000 9219
1	0000 0000 0000 0112	00	0000 0000 0001 2432
2	0000 0000 0000 0224	00	0000 0000 0002 4864
3	0000 0000 0000 0448	00	0000 0000 0004 90C8
.	.		.
.	.		.
.	.		.
34	0890 0000 0000 0000	09	2190 0000 0000 0000
35	1120 0000 0000 0000	12	4320 0000 0000 0000
36	2240 0000 0000 0000	24	8640 0000 0000 0000
37	4480 0000 0000 0000	49	0C80 0000 0000 0000

Section 2, Subsection 13

This subsection tests Multiply Partial Summing network for summing of a partial carry from the A Product network with a partial sum from the B Product network.

Hardware Tested:

- Multiply Partial Summing network.

Method:

Testing of the Multiply Partial Summing network for summing of partial carries from the A Product network with partial sums from the B Product network is done using the hardware controls described in the Section Method.

The multiplier for this subsection, 0000 0000 0022 0800(HEX), pulls the 1x decode from C register bits 65 through 67 and 61 through 63 into the A Product network and C register bits 71 through 73 into the B Product network. These decodes along with the multiplicand (test pattern) will cause a specific A Product network carry to be summed with a specific B Product network sum.

The test patterns start with the summing of MPC1A bit 65 with MPS1B bit 60 and walk through the summing of MPC1A bit 14 with MPS1B bit 9.

Initialization:

- Write 0000 0000 0022 0800(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
  a. Report test pattern; informational.

C.2 Test MPC1A and MPS1B summing using nonfeedback multiply.
  a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
  b. Wait halt.
  c. Read and report contents of RF0A (MSUM bits 8 through 71).
  d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
  a. Error if not equal to expected result.

C.4 Read/format and report status according to the Section Status Format.
  a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
  a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 in bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0811	00	0000 0000 0022 5051
1	0000 0000 0000 1022	00	0000 0000 0044 A0A2
2	0000 0000 0000 2044	00	0000 0000 0089 4144
3	0000 0000 0000 4088	00	0000 0000 0112 8288
.	.		.
.	.		.
.	.		.
30	0811 0000 0000 0000	22	5051 0000 0000 0000
31	1022 0000 0000 0000	44	A0A2 0000 0000 0000
32	2044 0000 0000 0000	89	4144 0000 0000 0000
33	4088 0000 0000 0000	12	8288 0000 0000 0000

NOTE

When pattern 33 is run, the result from the Multiply Partial Summing network is too large for the output path and the top bit is not output.

Section 2, Subsection 14

This subsection tests Multiply Partial Summing network for partial carry generation and the Multiply Final Adder for adding of partial carry values.

Hardware Tested:

- Multiply Partial Summing network.
- Multiply Final Adder.

Method:

Testing of the Multiply Partial Summing network for partial carry generation and looping on feedback along with the adders addition using partial carry inputs is done using the hardware controls described in the Section Method.

Two multiplies will be performed in this subsection. The first generates the PCR2 bit and then outputs it directly. The second generates the PCR2 bit, then feeds it back into the Multiply Partial Summing network where it is again summed and the output taken.

The test patterns start with the generation of PCR2 bit 68 and walk through the generation of PCR2 bit 12. The looping of the PCR2 bit will first show up at pattern number 4.

Initialization:

- Write multiplier of 0000 0000 0002 0800(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
  a. Report test pattern; informational.

C.2 Test PCR2 generation without feedback.
  a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
  b. Wait halt.
  c. Read and report contents of RF0A (MSUM bits 8 through 71).
  d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
  a. Error if not equal to expected result.

C.4 Read/format and report status according to the Section Status Format.
  a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test PCR2 looping using feedback.
  a. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
  b. Wait halt.
  c. Read and report contents of RF0A (MSUM bits 8 through 71).
  d. Error if not equal to expected result.

C.6 Read and report contents of RF09 bits 48 through 55 (MSUM bits 0 through 7).
  a. Error if not equal to expected result.

Conditions:

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2 and 5 report bits 8 through 71 in bits 0 through 63. Conditions 3 and 6 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0043	C.2 and C.3	00	0000 0000 0000 1103
		C.5 and C.6	00	0000 0000 0000 0001
1	0000 0000 0000 0086	C.2 and C.3	00	0000 0000 0000 2206
		C.5 and C.6	00	0000 0000 0000 0002
2	0000 0000 0000 010C	C.2 and C.3	00	0000 0000 0000 440C
		C.5 and C.6	00	0000 0000 0000 0004
3	0000 0000 0000 0218	C.2 and C.3	00	0000 0000 0000 8818
		C.5 and C.6	00	0000 0000 0000 0008
4	0000 0000 0000 0430	C.2 and C.3	00	0000 0000 0001 1030
		C.5 and C.6	00	0000 0000 0000 0011
5	0000 0000 0000 0860	C.2 and C.3	00	0000 0000 0002 2060
		C.5 and C.6	00	0000 0000 0000 0022
6	0000 0000 0000 10C0	C.2 and C.3	00	0000 0000 0004 40C0
		C.5 and C.6	00	0000 0000 0000 0044
7	0000 0000 0000 2180	C.2 and C.3	00	0000 0000 0008 8180
		C.5 and C.6	00	0000 0000 0000 0088
8	0000 0000 0000 4300	C.2 and C.3	00	0000 0000 0011 0300
		C.5 and C.6	00	0000 0000 0000 0110

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
9	0000 0000 0000 8600	C.2 and C.3	00	0000 0000 0022 0600
		C.5 and C.6	00	0000 0000 0000 0220
A	0000 0000 0001 0C00	C.2 and C.3	00	0000 0000 0044 0C00
		C.5 and C.6	00	0000 0000 0000 0440
B	0000 0000 0002 1800	C.2 and C.3	00	0000 0000 0088 1800
		C.5 and C.6	00	0000 0000 0000 0881
C	0000 0000 0004 3000	C.2 and C.3	00	0000 0000 0110 3000
		C.5 and C.6	00	0000 0000 0000 1103
D	0000 0000 0008 6000	C.2 and C.3	00	0000 0000 0220 6000
		C.5 and C.6	00	0000 0000 0000 2206
E	0000 0000 0010 C000	C.2 and C.3	00	0000 0000 0440 C000
		C.5 and C.6	00	0000 0000 0000 440C
F	0000 0000 0021 800	C.2 and C.3	00	0000 0000 0881 8000
		C.5 and C.6	00	0000 0000 0000 8818
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
36	10C0 0000 0000 0000	C.2 and C.3	04	40C0 0000 0000 0000
		C.5 and C.6	00	0044 0C00 0000 0000
37	2180 0000 0000 0000	C.2 and C.3	08	8180 0000 0000 0000
		C.5 and C.6	00	0088 1800 0000 0000
38	4300 0000 0000 0000	C.2 and C.3	11	0300 0000 0000 0000
		C.5 and C.6	00	0110 3000 0000 0000

Section 2, Subsection 15

This subsection tests the Multiply Partial Summing network for generation of N+2 carry.

Hardware Tested:

- Multiply Partial Summing network.

Method:

Testing of the Multiply Partial Summing network for N+2 carry generation is done using the hardware controls described in the Section Method.

Carry to N+2 is generated when four or more input bits are present at the same bit position of the Multiply Partial Summing network. When the carry must pass over array boundaries, it is done via the CROT signals. A carry to the first bit of the next array will go out the CROT (2) signal and a carry to the third bit of the next array will go out the CROT (0) signal.

The multiplier and the test pattern will cause only one N+2 carry to be generated at a time. The test patterns start with the summing of MPS1A bit 66, MPC1A bit 67, MPS1B bit 62, and MPC1B bit 63 and walk through the summing of MPS1A bit 13, MPC1A bit 14, MPS1B bit 9, and MPC1B bit 10.

Initialization:

- Write multiplier of 0000 0000 000A 2800(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test carry to N+2 generation using nonfeedback multiply.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to the Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 in bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 034D	00	0000 0000 0004 30C1
1	0000 0000 0000 069A	00	0000 0000 0008 6182
2	0000 0000 0000 0D34	00	0000 0000 0010 C304
3	0000 0000 0000 1A68	00	0000 0000 0021 8608
	.		.
	.		.
	.		.
	.		.
	.		.
31	0D34 0000 0000 0000	10	C304 0000 0000 0000
32	1A68 0000 0000 0000	21	8608 0000 0000 0000
33	34D0 0000 0000 0000	43	0C10 0000 0000 0000
34	69A0 0000 0000 0000	86	1820 0000 0000 0000

Section 2, Subsection 16

This subsection tests Multiply Partial Summing network for summing of a Multiply Partial Sum register bit with a Net B partial carry bit.

Hardware Tested:

- Multiply Partial Summing network.

Method:

The testing of the Multiply Partial Summing network for summing of Multiply Partial Sum register with the Net B partial carries is performed using the hardware controls described in the Section Method.

This subsection will generate a partial sum in PSR2 and then loop it back to be summed with a partial carry from the B Product network. The test patterns start with the summing of PSR2 bit 60 with MPC1B bit 69 and walk through the summing of PSR2 bit 13 with MPC1B bit 22.

Initialization:

- Write multiplier of 0000 0000 0280 0800(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test summing of PSR2 with MPC1B.
a. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 in bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 8005	00	0000 0000 0002 8021
1	0000 0000 0001 000A	00	0000 0000 0005 0042
2	0000 0000 0002 0014	00	0000 0000 000A 0084
3	0000 0000 0004 0028	00	0000 0000 0014 0108
4	0000 0000 0008 0050	00	0000 0000 0028 0210
5	0000 0000 0010 00A0	00	0000 0000 0050 0420
6	0000 0000 0020 0140	00	0000 0000 00A0 0840
7	0000 0000 0040 0280	00	0000 0000 0140 1080
8	0000 0000 0080 0500	00	0000 0000 0280 2100
9	0000 0000 0100 0A00	00	0000 0000 0500 4200
A	0000 0000 0200 1400	00	0000 0000 0A00 8401
B	0000 0000 0400 2800	00	0000 0000 1401 0802
C	0000 0000 0800 5000	00	0000 0000 2802 1005
Start of Sequential Pattern for Expected Result:			
D	0000 0000 1000 A000	00	0000 0000 5004 200A
E	0000 0000 2001 4000	00	0000 0000 A008 4014
F	0000 0000 4002 8000	00	0000 0001 4010 8028
10	0000 0000 8005 0000	00	0000 0002 8021 0050
	.		.
	.		.
	.		.
	.		.
2C	0800 5000 0000 0000	00	2802 1005 0000 0000
2D	1000 A000 0000 0000	00	5004 200A 0000 0000
2E	2001 4000 0000 0000	00	A008 4014 0000 0000
2F	4002 8000 0000 0000	01	4010 8028 0000 0000

Section 2, Subsection 17

This subsection tests Multiply Partial Summing network for summing of a multiply partial carry bit with a Net B partial sum bit.

Hardware Tested:

- Multiply Partial Summing network.

Method:

Testing of the Multiply Partial Summing network for summing partial carry with Net B partial sums is done using the hardware controls described in the Section Method.

This subsection will generate a partial carry in PCR2 and loop it back to be summed with a partial sum from the B Product network. The test patterns start with the summing of PCR2 bit 64 with MPS1B bit 71 and walk through the summing of PCR2 bit 12 with MPS1B bit 19.

Initialization:

- Write multiplier of 0000 0000 0080 2800(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test summing of PCR2 with MPS1B.
a. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 in bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0681	00	0000 0000 0000 0683
1	0000 0000 0000 0D02	00	0000 0000 0000 0D06
2	0000 0000 0000 1A04	00	0000 0000 0000 1A0C
3	0000 0000 0000 3408	00	0000 0000 0000 3418
4	0000 0000 0000 6810	00	0000 0000 0000 6830
5	0000 0000 0000 D020	00	0000 0000 0000 D061
6	0000 0000 0001 A040	00	0000 0000 0001 A0C2
7	0000 0000 0003 4080	00	0000 0000 0003 4184
8	0000 0000 0006 8100	00	0000 0000 0006 8308
9	0000 0000 000D 0200	00	0000 0000 000D 0610
A	0000 0000 001A 0400	00	0000 0000 001A 0C21
B	0000 0000 0034 0800	00	0000 0000 0034 1842
Start of Sequential Pattern for Expected Result:			
C	0000 0000 0068 1000	00	0000 0000 0068 3085
D	0000 0000 00D0 2000	00	0000 0000 00D0 610A
E	0000 0000 01A9 4000	00	0000 0000 01A0 C214

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
F	0000 0000 0340 8000	00	0000 0000 0341 8428
	.		.
	.		.
	.		.
	.		.
31	0D02 0000 0000 0000	00	0D06 10A0 0000 0000
32	1A04 0000 0000 0000	00	1A0C 2140 0000 0000
33	3408 0000 0000 0000	00	3418 4280 0000 0000
34	6810 0000 0000 0000	00	6830 8500 0000 0000

Section 2, Subsection 18

This subsection tests 1x, -2x, and complement correction decodes from C register bits 65 through 73 to the B and A Product networks.

Hardware Tested:

- Multiplier decoders for C register bits 67 through 73 to B Product network.
- Multiply decoder for C register bits 65 through 67 to A Product network.
- Multiply Partial Summing network for summing of correction bits.
- B register sign.
- A and B Product network.
- Multiply Final Adder.

Method:

The testing of the multiplier decoding for 1x, -2x, and complement correction from C register bits 67 through 73 is done using the hardware controls described in the Section Method.

The multiplier is changed for each condition that executes a micrand sequence. The multipliers used in this subsection should cause the following decodes by condition.

Condition 2: 1x decode from C register bits 71 through 73 to B Product network.

Condition 5: -2x and CCOR (11) from C register bits 71 through 73, and 1x from C register bits 69 through 71 to the B Product network.

Condition 8: -2x and CCOR (9) from C register bits 69 through 71, and 1x from C register bits 67 through 69 to the B Product network.

Condition 11: -2x and CCOR (7) from C register bits 67 through 69 to the B Product network, and 1x from C register bits 65 through 67 to the A Product network.

The last two test patterns run have negative signs and will cause all decodes to change their signs; a 0x becomes a -0x, a 1x becomes a -1x, a 2x becomes a -2x, and a -2x becomes a 2x. When the minus decode is pulled, the CCOR signal is generated. These last two patterns will test the CCOR (5, 3, and 1) signals and their summing in the Multiply Partial Summing network. They will also test the minus decodes from the C register bit groups 65 through 67, 63 through 65, and 61 through 63 and their use in the A Product network.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test 1x multiplier decode from C register bits 71 through 73 to the B Product network.
a. Write multiplier, 0000 0000 0000 0400(HEX), in RF0C.
b. Start processor at Micrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test 1x and -2x multiplier decodes from C register bits 69 through 73 to the B Product network.
a. Write multiplier, 0000 0000 0000 1000(HEX), in RF0C.
b. Start processor at Micrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

Conditions:

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test 1x and -2x multiplier decodes from C register bits 67 through 71 to the B Product network.
a. Write multiplier, 0000 0000 0000 4000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Test 1x and -2x multiplier decodes from C register bits 65 through 69 to the A and B Product networks.
a. Write multiplier, 0000 0000 0001 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.12 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.13 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.14 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, 8, and 11 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, 9, and 12 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
		C.11 and C.12	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	4AAA AAAA AAAA AAAA
		C.5 and C.6	00	9555 5555 5555 5554
		C.8 and C.9	02	5555 5555 5555 5550
		C.11 and C.12	09	5555 5555 5555 5540
2	5555 5555 5555 5555	C.2 and C.3	00	5555 5555 5555 5555
		C.5 and C.6	00	AAAA AAAA AAAA AAAA
		C.8 and C.9	02	AAAA AAAA AAAA AAA8
		C.11 and C.12	0A	AAAA AAAA AAAA AAA0
3	7FFF FFFF FFFF FFFF	C.2 and C.3	00	7FFF FFFF FFFF FFFF
		C.5 and C.6	00	FFFF FFFF FFFF FFFE
		C.8 and C.9	03	FFFF FFFF FFFF FFF8
		C.11 and C.12	0F	FFFF FFFF FFFF FFE0
4	3333 3333 3333 3333	C.2 and C.3	00	3333 3333 3333 3333
		C.5 and C.6	00	6666 6666 6666 6666
		C.8 and C.9	01	9999 9999 9999 9998
		C.11 and C.12	06	6666 6666 6666 6660
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	4CCC CCCC CCCC CCCC
		C.5 and C.6	00	9999 9999 9999 9998
		C.8 and C.9	02	6666 6666 6666 6660
		C.11 and C.12	09	9999 9999 9999 9980

NOTE

The following two patterns are used to check complement control based on B sign.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
6	AAAA AAAA AAAA AAAA	C.2 and C.3	00	5555 5555 5555 5556
		C.5 and C.6	00	AAAA AAAA AAAA AAAC
		C.8 and C.9	02	AAAA AAAA AAAA AAB0
		C.11 and C.12	0A	AAAA AAAA AAAA AAC0
7	C555 5555 5555 5555	C.2 and C.3	00	3AAA AAAA AAAA AAAB
		C.5 and C.6	00	7555 5555 5555 5556
		C.8 and C.9	01	D555 5555 5555 5558
		C.11 and C.12	07	5555 5555 5555 5560

Section 2, Subsection 19

This subsection tests 1x, -2x, and 2s correction decodes from C register bits 65 through 73 to the B and A Product networks and the upper Multiply Partial Summing network.

Hardware Tested:

- Multiplier decoders for C register bits 67 through 73 to B Product network.
- Multiply decoder for C register bits 65 through 67 to A Product network.
- Multiply Partial Summing network for summing of correction bits.
- B register sign.
- A and B Product network.
- Multiply Partial Summing network upper bits.
- Multiply Final Adder.
- Multiply Final Adder carry out of 12 bit iteration.

Method:

Testing of the multiplier decoding for 1x, -2x, and complement correction from C register bits 67 through 73 is performed using the hardware controls described in the Section Method.

The multiplier is changed for each condition that executes a micrand sequence. The multipliers used in this subsection should cause the following decodes by condition.

Condition 2: 1x decode from C register bits 71 through 73 to B Product network.

Condition 5: -2x and CCOR (11) from C register bits 71 through 73, and 1x from C register bits 69 through 71 to the B Product network.

Condition 8: -2x and CCOR (9) from C register bits 69 through 71, and 1x from C register bits 67 through 69 to the B Product network.

Condition 11: -2x and CCOR (7) from C register bits 67 through 69 to the B Product network, and 1x from C register bits 65 through 67 to the A Product network.

The last two test patterns run have negative signs and will cause all decodes to change their signs; a 0x becomes a -0x, a 1x becomes a -1x, a 2x becomes a -2x, and a -2x becomes a 2x. When the minus decode is pulled, the CCOR signal is generated. These last two patterns will test the CCOR (5, 3, and 1) signals and their summing in the Multiply Partial Summing network. They will also test the minus decodes from the C register bit groups 65 through 67, 63 through 65, and 61 through 63 and their use in the A Product network.

The partial sum and carry results of the first multiply are looped back in the Multiply Partial Summing network and resummed in the next multiply. This allows the results from the upper Multiply Partial Summing network to be output so that they can be checked.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test 1x Multiplier decodes from C register bits 71 through 73 to the B Product network and upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0000 0400, in RF0C.
b. Start processor at Micrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

Conditions:

C.5 Test 1x and -2x Multiplier decodes from C register bits 69 through 73 to the B Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0000 1000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test 1x and -2x Multiplier decodes from C register bits 67 through 71 to the B Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0000 4000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Test 1x and -2x Multiplier Decodes from C register bits 65 through 69 to the A and B Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0001 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.12 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.13 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.14 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, 8, and 11 report bits 8 through 71 in bits 0 through 63.
Conditions 3, 6, 9, and 12 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
		C.11 and C.12	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	0004 AAAA AAAA AAAA
		C.5 and C.6	00	0009 5555 5555 5555
		C.8 and C.9	00	0025 5555 5555 5555
		C.11 and C.12	00	0095 5555 5555 5555
2	5555 5555 5555 5555	C.2 and C.3	00	0005 5555 5555 5555
		C.5 and C.6	00	000A AAAA AAAA AAAA
		C.8 and C.9	00	002A AAAA AAAA AAAA
		C.11 and C.12	00	00AA AAAA AAAA AAAA
3	7FFF FFFF FFFF FFFF	C.2 and C.3	00	0007 FFFF FFFF FFFF
		C.5 and C.6	00	000F FFFF FFFF FFFF
		C.8 and C.9	00	003F FFFF FFFF FFFF
		C.11 and C.12	00	00FF FFFF FFFF FFFF
4	3333 3333 3333 3333	C.2 and C.3	00	0003 3333 3333 3333
		C.5 and C.6	00	0006 6666 6666 6666
		C.8 and C.9	00	0019 9999 9999 9999
		C.11 and C.12	00	0066 6666 6666 6666
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	0004 CCCC CCCC CCCC
		C.5 and C.6	00	0009 9999 9999 9999
		C.8 and C.9	00	0026 6666 6666 6666
		C.11 and C.12	00	0099 9999 9999 9999

NOTE

The following two patterns are used to check complement control based on B sign.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
6	AAAA AAAA AAAA AAAA	C.2 and C.3	00	0005 5555 5555 5555
		C.5 and C.6	00	000A AAAA AAAA AAAA
		C.8 and C.9	00	002A AAAA AAAA AAAA
		C.11 and C.12	00	00AA AAAA AAAA AAAA
7	C555 5555 5555 5555	C.2 and C.3	00	0003 AAAA AAAA AAAA
		C.5 and C.6	00	0007 5555 5555 5555
		C.8 and C.9	00	001D 5555 5555 5555
		C.11 and C.12	00	0075 5555 5555 5555

Section 2, Subsection 20

This subsection tests 1x, -2x, and complement correction decodes from C register bits 61 through 67 to the A Product network.

Hardware Tested:

- Multiplier decoder for C register bits 61 through 67 to the A Product network.
- A Product network.
- Multiply Final Adder.

Method:

The testing of multiplier decoding for 1x, -2x, and complement correction from C register bits 61 through 67 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection will be decoded as the following controls to the A Product network and CCOR signals to the Multiply Partial Summing network.

Condition 2: -2x and CCOR (5) from C register bits 65 through 67, and 1x from C register bits 63 through 65.

Condition 5: -2x and CCOR (3) from C register bits 63 through 65, and 1x from C register bits 61 through 63.

Condition 8: -2x and CCOR (1) from C register bits 61 through 63.

The negative sign of the last two test patterns will complement all the positive or negative controls from all C register decodes to both A and B Product networks.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test 1x and -2x Multiplier decodes from C register bits 63 through 67 to the A Product network.
a. Write multiplier, 0000 0000 0004 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test 1x and -2x Multiplier decodes from C register bits 61 through 63 to the A Product network.
a. Write multiplier, 0000 0000 0010 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test -2x Multiplier Decode from C register bits 61 through 63 to the A Product network.
a. Write multiplier, 0000 0000 0040 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

Conditions:

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, 8, and 11 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, 9, and 12 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	25	5555 5555 5555 5500
		C.5 and C.6	95	5555 5555 5555 5400
		C.8 and C.9	AA	AAAA AAAA AAAA B000
2	5555 5555 5555 5555	C.2 and C.3	2A	AAAA AAAA AAAA AA80
		C.5 and C.6	AA	AAAA AAAA AAAA AA00
		C.8 and C.9	55	5555 5555 5555 5800
3	7FFF FFFF FFFF FFFF	C.2 and C.3	3F	FFFF FFFF FFFF FF80
		C.5 and C.6	FF	FFFF FFFF FFFF FE00
		C.8 and C.9	00	0000 0000 0000 0800
4	3333 3333 3333 3333	C.2 and C.3	19	9999 9999 9999 9980
		C.5 and C.6	66	6666 6666 6666 6600
		C.8 and C.9	66	6666 6666 6666 6800

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
5	4CCC CCCC CCCC CCCC	C.2 and C.3	26	6666 6666 6666 6600
		C.5 and C.6	99	9999 9999 9999 9800
		C.8 and C.9	99	9999 9999 9999 A000

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	2A	AAAA AAAA AAAA AB00
		C.5 and C.6	AA	AAAA AAAA AAAA AC00
		C.8 and C.9	55	5555 5555 5555 5000
7	C555 5555 5555 5555	C.2 and C.3	1D	5555 5555 5555 5580
		C.5 and C.6	75	5555 5555 5555 5600
		C.8 and C.9	2A	AAAA AAAA AAAA A800

Section 2, Subsection 21

This subsection tests the 1x, -2x, and complement correction decodes from C register bits 61 through 67 to the A Product network and the upper Multiply Partial Summing network.

Hardware Tested:

- Multiply Partial Summing network upper bits.
- Multiplier decoder for C register bits 61 through 67 to the A Product network.
- A Product network.
- Multiply Final Adder.

Method:

The testing of multiplier decoding for 1x, -2x, and complement correction from C register bits 61 through 67 is done using the hardware controls described next.

The multipliers used in this subsection will be decoded as the following controls to the A Product network and CCOR signals to the Multiply Partial Summing network.

Condition 2: -2x and CCOR (5) from C register bits 65 through 67, and 1x from C register bits 63 through 65.

Condition 5: -2x and CCOR (3) from C register bits 63 through 65, and 1x from C register bits 61 through 63.

Condition 8: -2x and CCOR (1) from C register bits 61 through 63.

The negative sign of the last two test patterns will complement all the positive or negative controls from all C register decodes to both A and B Product networks.

The partial sum and carry results of the first multiply are looped back to the Multiply Partial Summing network and resummed in the next multiply. This allows the results from the upper Multiply Partial Summing network to be output so that they can be checked.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test the 1x and -2x multiplier decodes from C register bits 63 through 67 to the A Product network and upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0004 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test the 1x and -2x multiplier decodes from C register bits 61 through 63 to the A Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0010 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

Conditions:

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test -2x multiplier decode from C register bits 61 through 63 to the A Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0040 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63.
Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	0255 5555 5555 5555
		C.5 and C.6	00	0955 5555 5555 5555
		C.8 and C.9	00	2555 5555 5555 5555
2	5555 5555 5555 5555	C.2 and C.3	00	02AA AAAA AAAA AAAA
		C.5 and C.6	00	0AAA AAAA AAAA AAAA
		C.8 and C.9	00	2AAA AAAA AAAA AAAA
3	7FFF FFFF FFFF FFFF	C.2 and C.3	00	03FF FFFF FFFF FFFF
		C.5 and C.6	00	0FFF FFFF FFFF FFFF
		C.8 and C.9	00	3FFF FFFF FFFF FFFF
4	3333 3333 3333 3333	C.2 and C.3	00	0199 9999 9999 9999
		C.5 and C.6	00	0666 6666 6666 6666
		C.8 and C.9	00	1999 9999 9999 9999
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	0266 6666 6666 6666
		C.5 and C.6	00	0999 9999 9999 9999
		C.8 and C.9	00	2666 6666 6666 6666

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	00	02AA AAAA AAAA AAAA
		C.5 and C.6	00	0AAA AAAA AAAA AAAA
		C.8 and C.9	00	2AAA AAAA AAAA AAAB
7	C555 5555 5555 5555	C.2 and C.3	00	01D5 5555 5555 5555
		C.5 and C.6	00	0755 5555 5555 5555
		C.8 and C.9	00	1D55 5555 5555 5555

Section 2, Subsection 22

This subsection tests the 2x decodes from C register bits 67 through 73 to the B Product network.

Hardware Tested:

- Multiplier decoder for C register bits 67 through 73.
- B Product network.
- Multiply Final Adder.

Method:

The testing of the multiplier decoding for 2x in C register bits 67 through 73 is performed using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the B Product network and CCOR signals to the Multiply Partial Summing network.

Condition 2: 2x from C register bits 71 through 73.

Condition 5: 2x from C register bits 69 through 71, and -2x and CCOR (11) from C register bits 71 through 73.

Condition 8: 2x from C register bits 67 through 69, and -2x and CCOR (9) from C register bits 69 through 71.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test 2x multiplier decodes from C register bits 71 through 73 to the B Product network.
a. Write multiplier, 0000 0000 0000 0C00(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

Conditions:

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test 2x multiplier decodes from C register bits 69 through 71 to the B Product network.
a. Write multiplier, 0000 0000 0000 3000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test 2x multiplier decode from C register bits 67 through 69 to the B Product network.
a. Write multiplier, 0000 0000 0000 C000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	9555 5555 5555 5554
		C.5 and C.6	01	BFFF FFFF FFFF FFFC
		C.8 and C.9	06	FFFF FFFF FFFF FFF0
2	5555 5555 5555 5555	C.2 and C.3	00	AAAA AAAA AAAA AAAA
		C.5 and C.6	01	FFFF FFFF FFFF FFFE
		C.8 and C.9	07	FFFF FFFF FFFF FFF8
3	7FFF FFFF FFFF FFFF	C.2 and C.3	00	FFFF FFFF FFFF FFFE
		C.5 and C.6	02	FFFF FFFF FFFF FFFA
		C.8 and C.9	0B	FFFF FFFF FFFF FFE8
4	3333 3333 3333 3333	C.2 and C.3	00	6666 6666 6666 6666
		C.5 and C.6	01	3333 3333 3333 3332
		C.8 and C.9	04	CCCC CCCC CCCC CCC8
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	9999 9999 9999 9998
		C.5 and C.6	01	CCCC CCCC CCCC CCC8
		C.8 and C.9	07	3333 3333 3333 3320

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	00	AAAA AAAA AAAA AAAC
		C.5 and C.6	02	0000 0000 0000 0004
		C.8 and C.9	08	0000 0000 0000 0010
7	C555 5555 5555 5555	C.2 and C.3	00	7555 5555 5555 5556
		C.5 and C.6	01	8000 0000 0000 0002
		C.8 and C.9	05	8000 0000 0000 0008

Section 2, Subsection 23

This subsection tests the 2x decodes from C register bits 67 through 73 to the B Product network and the upper bits of the Multiply Partial Summing network.

Hardware Tested:

- Multiply Partial Summing network upper bits.
- Multiplier decoder for C register bits 67 through 73.
- B Product network.
- Multiply Final Adder.

Method:

The testing of the multiplier decoding for 2x in C register bits 67 through 73 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the B Product network and CCOR signals to the Multiply Partial Summing network.

Condition 2: 2x from C register bits 71 through 73.

Condition 5: 2x from C register bits 69 through 71, and -2x and CCOR (11) from C register bits 71 through 73.

Condition 8: 2x from C register bits 67 through 69, and -2x and CCOR (9) from C register bits 69 through 71.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

The partial sum and carry results of the first multiply are looped back in the Multiply Partial Summing network and resummed in the next multiply. This allows the results from the upper Multiply Partial Summing network to be output so that they can be checked.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

Conditions:

C.2 Test the 2x multiplier decodes from C register bits 71 through 73 to the B Product network and upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0000 0C00(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test 2x multiplier decodes from C register bits 69 through 71 to the B Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0000 3000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test 2x multiplier decodes from C register bits 67 through 69 to the B Product network and the upper Multiply Partial Summing network.
a. Write Multiplier, 0000 0000 0000 C000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	0009 5555 5555 5555
		C.5 and C.6	00	001B FFFF FFFF FFFF
		C.8 and C.9	00	006F FFFF FFFF FFFF
2	5555 5555 5555 5555	C.2 and C.3	00	000A AAAA AAAA AAAA
		C.5 and C.6	00	001F FFFF FFFF FFFF
		C.8 and C.9	00	007F FFFF FFFF FFFF
3	7FFF FFFF FFFF FFFF	C.2 and C.3	00	000F FFFF FFFF FFFF
		C.5 and C.6	00	002F FFFF FFFF FFFF
		C.8 and C.9	00	00BF FFFF FFFF FFFF
4	3333 3333 3333 3333	C.2 and C.3	00	0006 6666 6666 6666
		C.5 and C.6	00	0013 3333 3333 3333
		C.8 and C.9	00	004C CCCC CCCC CCCC
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	0009 9999 9999 9999
		C.5 and C.6	00	001C CCCC CCCC CCCC
		C.8 and C.9	00	0073 3333 3333 3333

NOTE

The following two patterns are used to check complement control based on B sign.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
6	AAAA AAAA AAAA AAAA	C.2 and C.3	00	000A AAAA AAAA AAAA
		C.5 and C.6	00	0020 0000 0000 0000
		C.8 and C.9	00	0080 0000 0000 0000
7	C555 5555 5555 5555	C.2 and C.3	00	0007 5555 5555 5555
		C.5 and C.6	00	0016 0000 0000 0000
		C.8 and C.9	00	0058 0000 0000 0000

Section 2, Subsection 24

This subsection tests 2x decodes from C register bits 61 through 67 to the A Product network.

Hardware Tested:

- Multiplier decoder for C register bits 61 through 67.
- A Product network.
- Multiply Final Adder.

Method:

The testing of the multiplier decoding for 2x in C register bits 61 through 67 is performed using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the A Product network and CCOR signals to the Multiply Partial Summing network.

Method:

Condition 2: 2x from C register bits 65 through 67. Also, -2x and CORR (7) from C register bits 67 through 69 are generated to the B Product network.

Condition 5: 2x from C register bits 63 through 65, and -2x and CCOR (5) from C register bits 65 through 67.

Condition 8: 2x from C register bits 61 through 63, and -2x and CCOR (3) from C register bits 63 through 65.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register controls to both the A and B Product networks.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test 2x multiplier decodes from C register bits 65 through 67 to the A Product network.
a. Write multiplier, 0000 0000 0003 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test 2x multiplier decodes from C register bits 63 through 65 to the A Product network.
a. Write multiplier, 0000 0000 000C 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test 2x multiplier decodes from C register bits 61 through 63 to the Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0030 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

Conditions:

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	1B	FFFF FFFF FFFF FFC0
		C.5 and C.6	6F	FFFF FFFF FFFF FF00
		C.8 and C.9	BF	FFFF FFFF FFFF FC00
2	5555 5555 5555 5555	C.2 and C.3	1F	FFFF FFFF FFFF FFE0
		C.5 and C.6	7F	FFFF FFFF FFFF FF80
		C.8 and C.9	FF	FFFF FFFF FFFF FE00
3	7FFF FFFF FFFF FFFF	C.2 and C.3	2F	FFFF FFFF FFFF FFA0
		C.5 and C.6	BF	FFFF FFFF FFFF FE80
		C.8 and C.9	FF	FFFF FFFF FFFF FA00
4	3333 3333 3333 3333	C.2 and C.3	13	3333 3333 3333 3320
		C.5 and C.6	4C	CCCC CCCC CCCC CC80
		C.8 and C.9	33	3333 3333 3333 3200
5	4CCC CCCC CCCC CCCC	C.2 and C.3	1C	CCCC CCCC CCCC CC80
		C.5 and C.6	73	3333 3333 3333 3200
		C.8 and C.9	CC	CCCC CCCC CCCC C800

NOTE

The following two patterns are used to check complement control based on B sign.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
6	AAAA AAAA AAAA AAAA	C.2 and C.3	20	0000 0000 0000 0040
		C.5 and C.6	80	0000 0000 0000 0100
		C.8 and C.9	00	0000 0000 0000 0400
7	C555 5555 5555 5555	C.2 and C.3	16	0000 0000 0000 0020
		C.5 and C.6	58	0000 0000 0000 0080
		C.8 and C.9	60	0000 0000 0000 0200

Section 2, Subsection 25

This subsection tests the 2x decodes from C register bits 61 through 67 to the A Product network and the upper bits of the Multiply Partial Summing network.

Hardware Tested:

- Multiply Partial Summing network upper bits.
- Multiplier decoder for C register bits 61 through 67.
- A Product network.
- Multiply Final Adder.

Method:

The testing of the multiplier decoding for 2x in C register bits 61 through 67 is performed using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the A Product network and CCOR signals to the Multiply Partial Summing network.

Condition 2: 2x from C register bits 65 through 67. Also, -2x and CORR (7) from C register bits 67 through 69 are generated to the B Product network.

Condition 5: 2x from C register bits 63 through 65, and -2x and CCOR (5) from C register bits 65 through 67.

Condition 8: 2x from C register bits 61 through 63, and -2x and CCOR (3) from C register bits 63 through 65.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

The partial sum and carry results of the first multiply are looped back in the Multiply Partial Summing network and resummed in the next multiply. This allows the results from the upper Multiply Partial Summing network to be output so that they can be checked.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test the 2x multiplier decodes from C register bits 65 through 67 to the A Product network and upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0003 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test the 2x multiplier decodes from C register bits 63 through 65 to the A Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 000C 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

Conditions:

C.8 Test the 2x multiplier decodes from C register bits 61 through 63 to the A Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0030 0000(HEX), in RF0C.
b. Start processor at Micrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	01BF FFFF FFFF FFFF
		C.5 and C.6	00	06FF FFFF FFFF FFFF
		C.8 and C.9	00	1BFF FFFF FFFF FFFF
2	5555 5555 5555 5555	C.2 and C.3	00	01FF FFFF FFFF FFFF
		C.5 and C.6	00	07FF FFFF FFFF FFFF
		C.8 and C.9	00	1FFF FFFF FFFF FFFF
3	7FFF FFFF FFFF FFFF	C.2 and C.3	00	02FF FFFF FFFF FFFF
		C.5 and C.6	00	0BFF FFFF FFFF FFFF
		C.8 and C.9	00	2FFF FFFF FFFF FFFF
4	3333 3333 3333 3333	C.2 and C.3	00	0133 3333 3333 3333
		C.5 and C.6	00	04CC CCCC CCCC CCCC
		C.8 and C.9	00	1333 3333 3333 3333
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	01CC CCCC CCCC CCCC
		C.5 and C.6	00	0733 3333 3333 3333
		C.8 and C.9	00	1CCC CCCC CCCC CCCC

NOTE

The following two patterns are used to check complement control based on B sign.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
6	AAAA AAAA AAAA AAAA	C.2 and C.3	00	0200 0000 0000 0000
		C.5 and C.6	00	0800 0000 0000 0000
		C.8 and C.9	00	2000 0000 0000 0000
7	C555 5555 5555 5555	C.2 and C.3	00	0160 0000 0000 0000
		C.5 and C.6	00	0580 0000 0000 0000
		C.8 and C.9	00	1600 0000 0000 0000

Section 2, Subsection 26

This subsection tests -1x decodes on a 101 binary pattern from C register bits 67 through 73 to the B Product network.

Hardware Tested:

- Multiplier decoders for C register bits 67 through 73.
- B Product network.
- Multiply Final Adder.

Method:

Testing of the multiplier decoding of -1x on a 101 binary pattern in C register bits 67 through 73 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the B Product network and the listed CCOR signals to the Multiply Partial Summing network.

Condition 2: -1x and CCOR (11) in C register bits 71 through 73, and 1x in C register bits 69 through 71.

Condition 5: -1x and CCOR (9) in C register bits 69 through 71, -2x and CCOR (11) in C register bits 71 through 73, and 1x in C register bits 67 through 69.

Condition 8: -1x and CCOR (7) in C register bits 65 through 67, and -2x and CCOR (9) in C register bits 67 through 69. Also, 1x in C register bits 65 through 67 to the A Product network.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test -1x multiplier decodes from C register bits 71 through 73 to the B Product network.
a. Write multiplier, 0000 0000 0000 1400(HEX), in RF0C.
b. Start processor at Micrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test -1x multiplier decodes from C register bits 69 through 71 to the B Product network.
a. Write multiplier, 0000 0000 0000 5000(HEX), in RF0C.
b. Start processor at Micrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

Conditions:

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test the -1x multiplier decodes from C register bits 67 through 69 to the B Product network.
a. Write multiplier, 0000 0000 0001 4000(HEX), in RF0C.
b. Start processor at Micrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	DFFF FFFF FFFF FFFE
		C.5 and C.6	02	EAAA AAAA AAAA AAA4
		C.8 and C.9	0B	AAAA AAAA AAAA AA90
2	5555 5555 5555 5555	C.2 and C.3	00	FFFF FFFF FFFF FFFF
		C.5 and C.6	03	5555 5555 5555 5552
		C.8 and C.9	0D	5555 5555 5555 5548
3	7FFF FFFF FFFF FFFF	C.2 and C.3	01	7FFF FFFF FFFF FFFD
		C.5 and C.6	04	FFFF FFFF FFFF FFF6
		C.8 and C.9	13	FFFF FFFF FFFF FFD8
4	3333 3333 3333 3333	C.2 and C.3	00	9999 9999 9999 9999
		C.5 and C.6	01	FFFF FFFF FFFF FFFE
		C.8 and C.9	07	FFFF FFFF FFFF FFF8
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	E666 6666 6666 6666
		C.5 and C.6	02	FFFF FFFF FFFF FFF8
		C.8 and C.9	0B	FFFF FFFF FFFF FFE0
	NOTE The following two patterns are used to check complement control based on B sign.			
6	AAAA AAAA AAAA AAAA	C.2 and C.3	01	0000 0000 0000 0002
		C.5 and C.6	03	5555 5555 5555 555C
		C.8 and C.9	0D	5555 5555 5555 5570
7	C555 5555 5555 5555	C.2 and C.3	00	B000 0000 0000 0001
		C.5 and C.6	02	4AAA AAAA AAAA AAAE
		C.8 and C.9	09	2AAA AAAA AAAA AAB8

Section 2, Subsection 27

This subsection tests the -1x decodes on a 101 binary pattern from C register bits 67 through 73 to the B Product network, and the upper bits of the Multiply Partial Summing network.

Hardware Tested:

- Multiply Partial Summing network upper bits.
- Multiplier decoders for C register bits 67 through 73.
- B Product network.
- Multiply Final Adder.

Method:

Testing of the multiplier decoding of -1x on a 101 binary pattern in C register bits 67 through 73 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the B Product network and CCOR signals to the Multiply Partial Summing network.

Condition 2: -1x and CCOR (11) in C register bits 71 through 73, and 1x in C register bits 69 through 71.

Condition 5: -1x and CCOR (9) in C register bits 69 through 71, -2x and CCOR (11) in C register bits 71 through 73, and 1x in C register bits 67 through 69.

Condition 8: -1x and CCO (7) in C register bits 65 through 67, and -2x and CCOR (9) in C register bits 67 through 69. Also, 1x in C register bits 65 through 67 to the A Product network.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

The partial sum and carry results of the first multiply are looped back to the Multiply Partial Summing network and resummed in the next multiply. This allows the results from the upper Multiply Partial Summing network to be output so that they can be checked.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test the -1x multiplier decodes from C register bits 71 through 73 to the B Product network and upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0000 1400(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

Conditions:

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test the -1x multiplier decodes from C register bits 69 through 71 to the B Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0000 5000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test the -1x multiplier decodes from C register bits 67 through 69 to the B Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0001 4000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	000D FFFF FFFF FFFF
		C.5 and C.6	00	002E AAAA AAAA AAAA
		C.8 and C.9	00	00BA AAAA AAAA AAAA
2	5555 5555 5555 5555	C.2 and C.3	00	000F FFFF FFFF FFFF
		C.5 and C.6	00	0035 5555 5555 5555
		C.8 and C.9	00	00D5 5555 5555 5555
3	7FFF FFFF FFFF FFFF	C.2 and C.3	00	0017 FFFF FFFF FFFF
		C.5 and C.6	00	004F FFFF FFFF FFFF
		C.8 and C.9	00	013F FFFF FFFF FFFF
4	3333 3333 3333 3333	C.2 and C.3	00	0009 9999 9999 9999
		C.5 and C.6	00	001F FFFF FFFF FFFF
		C.8 and C.9	00	007F FFFF FFFF FFFF
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	000E 6666 6666 6666
		C.5 and C.6	00	002F FFFF FFFF FFFF
		C.8 and C.9	00	00BF FFFF FFFF FFFF

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	00	0010 0000 0000 0000
		C.5 and C.6	00	0035 5555 5555 5555
		C.8 and C.9	00	00D5 5555 5555 5555
7	C555 5555 5555 5555	C.2 and C.3	00	000B 0000 0000 0000
		C.5 and C.6	00	0024 AAAA AAAA AAAA
		C.8 and C.9	00	0092 AAAA AAAA AAAA

Section 2, Subsection 28

This subsection tests -1x decodes on a 101 binary pattern from C register bits 61 through 67 to the A Product network.

Hardware Tested:

- Multiplier decoders for C register bits 61 through 67.
- A Product network.
- Multiply Final Adder.

Method:

The testing of the multiplier decoding of -1x on a 101 binary pattern in C register bits 61 through 67 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the A Product network and the listed CCOR signals to the Multiply Partial Summing network.

Method:

Condition 2: -1x and CCOR (5) in C register bits 65 through 67, and 1x in C register bits 63 through 65. Also, -2x and CCOR (7) in C register bits 67 through 69 to the B Product network.

Condition 5: -1x and CCOR (3) in C register bits 63 through 65, 1x in C register bits 61 through 63, and -2x and CCOR (5) in C register bits 65 through 67.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

Conditions:

C.2 Test -1x multiplier decodes from C register bits 65 through 67 to the A Product network.
a. Write multiplier, 0000 0000 0005 0000(HEX), in RF0C.
b. Start processor at Micrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test -1x multiplier decodes from C register bits 63 through 65 to the A Product network.
a. Write multiplier, 0000 0000 0014 0000(HEX), in RF0C.
b. Start processor at Micrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test -1x multiplier decodes from C register bits 61 through 63 to the A Product network.
a. Write multiplier, 0000 0000 0050 0000(HEX), in RF0C.
b. Start processor at Micrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.
Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	2E	AAAA AAAA AAAA AA40
		C.5 and C.6	BA	AAAA AAAA AAAA A900
		C.8 and C.9	40	0000 0000 0000 0400
2	5555 5555 5555 5555	C.2 and C.3	35	5555 5555 5555 5520
		C.5 and C.6	D5	5555 5555 5555 5480
		C.8 and C.9	00	0000 0000 0000 0200
3	7FFF FFFF FFFF FFFF	C.2 and C.3	4F	FFFF FFFF FFFF FF60
		C.5 and C.6	3F	FFFF FFFF FFFF FD80
		C.8 and C.9	00	0000 0000 0000 0600
4	3333 3333 3333 3333	C.2 and C.3	1F	FFFF FFFF FFFF FFE0
		C.5 and C.6	7F	FFFF FFFF FFFF FF80
		C.8 and C.9	CC	CCCC CCCC CCCC CE00
5	4CCC CCCC CCCC CCCC	C.2 and C.3	2F	FFFF FFFF FFFF FF80
		C.5 and C.6	BF	FFFF FFFF FFFF FE00
		C.8 and C.9	33	3333 3333 3333 3800

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	35	5555 5555 5555 55C0
		C.5 and C.6	D5	5555 5555 5555 5700
		C.8 and C.9	FF	FFFF FFFF FFFF FC00
7	C555 5555 5555 5555	C.2 and C.3	24	AAAA AAAA AAAA AAE0
		C.5 and C.6	92	AAAA AAAA AAAA AB80
		C.8 and C.9	9F	FFFF FFFF FFFF FE00

Section 2, Subsection 29

This subsection tests the -1x decodes on a 101 binary pattern from C register bits 61 through 67 to the A Product network and the upper bits of the Multiply Partial Summing network.

Hardware Tested:

- Multiply Partial Summing network upper bits.
- Multiplier decoders for C register bits 61 through 67.
- A Product network.
- Multiply Final Adder.

Method:

The testing of the multiplier decoding of -1x on a 101 binary pattern in C register bits 61 through 67 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the A Product network and the listed CCOR signals to the Multiply Partial Summing network.

Condition 2: -1x and CCOR (5) in C register bits 65 through 67, and 1x in C register bits 63 through 65. Also, -2x and CCOR (7) in C register bits 67 through 69 to the B Product network.

Condition 5: -1x and CCOR (3) in C register bits 63 through 65, 1x in C register bits 61 through 63, and -2x and CCOR (5) in C register bits 65 through 67.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

The partial sum and carry results of the first multiply are looped back to the Multiply Partial Summing network and resummed in the next multiply. This allows the results from the upper Multiply Partial Summing network to be output so that they can be checked.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

Conditions:

C.2 Test the -1x multiplier decodes from C register bits 65 through 67 to the A Product network and upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0005 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
b. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test the -1x multiplier decodes from C register bits 63 through 65 to the A Product network and the upper Multiply Partial Summing network.
c. Write multiplier, 0000 0000 0014 0000(HEX), in RF0C.
d. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
e. Wait halt.
f. Read and report contents of RF0A (MSUM bits 8 through 71).
g. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test the -1x multiplier decodes from C register bits 61 through 63 to the A Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0050 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

IF RI is set, go to C.2.
Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	02EA AAAA AAAA AAAA
		C.5 and C.6	00	0BAA AAAA AAAA AAAA
		C.8 and C.9	00	2EAA AAAA AAAA AAAA
2	5555 5555 5555 5555	C.2 and C.3	00	0355 5555 5555 5555
		C.5 and C.6	00	0D55 5555 5555 5555
		C.8 and C.9	00	3555 5555 5555 5555
3	7FFF FFFF FFFF FFFF	C.2 and C.3	00	04FF FFFF FFFF FFFF
		C.5 and C.6	00	13FF FFFF FFFF FFFF
		C.8 and C.9	00	4FFF FFFF FFFF FFFF
4	3333 3333 3333 3333	C.2 and C.3	00	01FF FFFF FFFF FFFF
		C.5 and C.6	00	07FF FFFF FFFF FFFF
		C.8 and C.9	00	1FFF FFFF FFFF FFFF
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	02FF FFFF FFFF FFFF
		C.5 and C.6	00	0BFF FFFF FFFF FFFF
		C.8 and C.9	00	2FFF FFFF FFFF FFFF

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	00	0355 5555 5555 5555
		C.5 and C.6	00	0D55 5555 5555 5555
		C.8 and C.9	00	3555 5555 5555 5555
7	C555 5555 5555 5555	C.2 and C.3	00	024A AAAA AAAA AAAA
		C.5 and C.6	00	092A AAAA AAAA AAAA
		C.8 and C.9	00	24AA AAAA AAAA AAAA

Section 2, Subsection 30

This subsection tests -1x decodes on a 110 binary pattern from C register bits 67 through 73 to the B Product network.

Hardware Tested:

- Multiplier decoders for C register bits 67 through 73.
- B Product network.
- Multiply Final Adder.

Method:

The testing of the multiplier decoding of -1x on a 110 binary pattern in C register bits 67 through 73 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the A Product network and the listed CCOR signals to Multiply Partial Summing network.

Condition 2: -1x and CCOR (11) in C register bits 71 through 73, and -2x and CCOR (9) in C register bits 69 through 71.

Condition 5: -1x and CCOR (9) in C register bits 69 through 71, and -2x and CCOR (7) in C register bits 65 through 67.

Condition 8: -1x and CCOR (7) in C register bits 67 through 69. Also, -2x and CCOR (5) in C register bits 65 through 67 to the A Product network.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

Conditions:

C.2 Test -1x multiplier decodes from C register bits 71 through 73 to the B Product network.
a. Write multiplier, 0000 0000 0000 1800(HEX), in RF0C.
b. Start processor at Microand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test -1x multiplier decodes from C register bits 69 through 71 to the B Product network.
a. Write multiplier, 0000 0000 0000 6000(HEX), in RF0C.
b. Start processor at Microand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test -1x multiplier decodes from C register bits 67 through 69 to the B Product network.
a. Write multiplier, 0000 0000 0001 8000(HEX), in RF0C.
b. Start processor at Microand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.
Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	DFFF FFFF FFFF FFFE
		C.5 and C.6	03	7FFF FFFF FFFF FFF8
		C.8 and C.9	0D	FFFF FFFF FFFF FFE0
2	5555 5555 5555 5555	C.2 and C.3	00	FFFF FFFF FFFF FFFF
		C.5 and C.6	03	FFFF FFFF FFFF FFFC
		C.8 and C.9	0F	FFFF FFFF FFFF FFD0
3	7FFF FFFF FFFF FFFF	C.2 and C.3	01	7FFF FFFF FFFF FFFD
		C.5 and C.6	05	FFFF FFFF FFFF FFF4
		C.8 and C.9	17	FFFF FFFF FFFF FFFD
4	3333 3333 3333 3333	C.2 and C.3	00	9999 9999 9999 9999
		C.5 and C.6	02	6666 6666 6666 6664
		C.8 and C.9	09	9999 9999 9999 9990
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	E666 6666 6666 6664
		C.5 and C.6	03	9999 9999 9999 9990
		C.8 and C.9	0E	6666 6666 6666 6640

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	01	0000 0000 0000 0002
		C.5 and C.6	04	0000 0000 0000 0008
		C.8 and C.9	10	0000 0000 0000 0020
7	C555 5555 5555 5555	C.2 and C.3	00	B000 0000 0000 0001
		C.5 and C.6	02	C000 0000 0000 0004
		C.8 and C.9	0B	0000 0000 0000 0010

Section 2, Subsection 31

This subsection tests the -1x decodes on a 110 binary pattern from C register bits 67 through 73 to the B Product network and the upper bits of the Multiply Partial Summing network.

Hardware Tested:

- Multiply Partial Summing network upper bits.
- Multiplier decoders for C register bits 67 through 73.
- B Product network.
- Multiply Final Adder.

Method:

The testing of the multiplier decoding of -1x on a 110 binary pattern in C register bits 67 through 73 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the B Product network and the listed CCOR signals to Multiply Partial Summing network.

Condition 2: -1x and CCOR (11) in C register bits 71 through 73, and -2x and CCOR (9) in C register bits 69 through 71.

Condition 5: -1x and CCOR (9) in C register bits 69 through 71, and -2x and CCOR (7) in C register bits 65 through 67.

Condition 8: -1x and CCOR (7) in C register bits 67 through 69. Also, -2x and CCOR (5) in C register bits 65 through 67 to the A Product network.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

The partial sum and carry results of the first multiply are looped back to the Multiply Partial Summing network and resummed in the next multiply. This allows the results from the upper Multiply Partial Summing network to be output so that they can be checked.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

Conditions:

C.2 Test the -1x multiplier decodes from C register bits 71 through 73 to the B Product network and upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0000 1800(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test the -1x multiplier decodes from C register bits 69 through 71 to the A Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0000 6000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test the -1x multiplier decodes from C register bits 67 through 69 to the B Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0001 8000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.
Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	000D FFFF FFFF FFFF
		C.5 and C.6	00	0037 FFFF FFFF FFFF
		C.8 and C.9	00	00DF FFFF FFFF FFFF
2	5555 5555 5555 5555	C.2 and C.3	00	000F FFFF FFFF FFFF
		C.5 and C.6	00	003F FFFF FFFF FFFF
		C.8 and C.9	00	00FF FFFF FFFF FFFF
3	7FFF FFFF FFFF FFFF	C.2 and C.3	00	0017 FFFF FFFF FFFF
		C.5 and C.6	00	005F FFFF FFFF FFFF
		C.8 and C.9	00	017F FFFF FFFF FFFF
4	3333 3333 3333 3333	C.2 and C.3	00	0009 9999 9999 9999
		C.5 and C.6	00	0026 6666 6666 6666
		C.8 and C.9	00	0099 9999 9999 9999
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	000E 6666 6666 6666
		C.5 and C.6	00	0039 9999 9999 9999
		C.8 and C.9	00	00E6 6666 6666 6666

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	00	0010 0000 0000 0000
		C.5 and C.6	00	0040 0000 0000 0000
		C.8 and C.9	00	0100 0000 0000 0000
7	C555 5555 5555 5555	C.2 and C.3	00	000B 0000 0000 0000
		C.5 and C.6	00	002C 0000 0000 0000
		C.8 and C.9	00	00B0 0000 0000 0000

Section 2, Subsection 32

This subsection tests -1x decodes on a 110 binary pattern from C register bits 61 through 67 to the A Product network.

Hardware Tested:

- Multiplier decoders for C register bits 61 through 67.
- A Product network.
- Multiply Final Adder.

Method:

The testing of the multiplier decoding of -1x on a 110 binary pattern in C register bits 61 through 67 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the A Product network and the listed CCOR signals to Multiply Partial Summing Network.

Condition 2: -1x and CCOR (5) in C register bits 65 through 67, and -2x and CCOR (3) in C register bits 63 through 65.

Condition 5: -1x and CCOR (3) in C register bits 63 through 65, and -2x and CCOR (1) in C register bits 61 through 63.

Condition 8: -1x and CCOR (7) in C register bits 67 through 69. Also, -2x and CCOR (1) in C register bits 61 through 63.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test -1x multiplier decodes from C register bits 65 through 67 to the A Product network.
a. Write multiplier, 0000 0000 0006 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

Conditions:

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test -1x multiplier decodes from C register bits 63 through 65 to the A Product network.
a. Write multiplier, 0000 0000 0018 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test -1x multiplier decodes from C register bits 61 through 63 to the A Product network.
a. Write multiplier, 0000 0000 0060 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	37	FFFF FFFF FFFF FF80
		C.5 and C.6	DF	FFFF FFFF FFFF FE00
		C.8 and C.9	D5	5555 5555 5555 5800
2	5555 5555 5555 5555	C.2 and C.3	3F	FFFF FFFF FFFF FFC0
		C.5 and C.6	FF	FFFF FFFF FFFF FF00
		C.8 and C.9	AA	AAAA AAAA AAAA AC00
3	7FFF FFFF FFFF FFFF	C.2 and C.3	5F	FFFF FFFF FFFF FF40
		C.5 and C.6	7F	FFFF FFFF FFFF FD00
		C.8 and C.9	00	0000 0000 0000 0400
4	3333 3333 3333 3333	C.2 and C.3	26	6666 6666 6666 6640
		C.5 and C.6	99	9999 9999 9999 9900
		C.8 and C.9	33	3333 3333 3333 3400
5	4CCC CCCC CCCC CCCC	C.2 and C.3	39	9999 9999 9999 9900
		C.5 and C.6	E6	6666 6666 6666 6400
		C.8 and C.9	CC	CCCC CCCC CCCC D000

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	40	0000 0000 0000 0080
		C.5 and C.6	00	0000 0000 0000 0200
		C.8 and C.9	AA	AAAA AAAA AAAA A800
7	C555 5555 5555 5555	C.2 and C.3	2C	0000 0000 0000 0040
		C.5 and C.6	B0	0000 0000 0000 0100
		C.8 and C.9	15	5555 5555 5555 5400

Section 2, Subsection 33

This subsection tests the -1x decodes on a 110 binary pattern in C register bits 61 through 67 to the A Product network and the upper bits of the Multiply Partial Summing network.

Hardware Tested:

- Multiply Partial Summing network upper bits.
- Multiplier decoders for C register bits 61 through 67.
- A Product network.
- Multiply Final Adder.

Method:

The testing of the multiplier decoding for -1x on a 110 binary pattern in C register bits 61 through 67 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the A Product network and the listed CCOR signals to Multiply Partial Summing network.

Condition 2: -1x and CCOR (5) in C register bits 65 through 67, and -2x and CCOR (3) in C register bits 63 through 65.

Condition 5: -1x and CCOR (3) in C register bits 63 through 65, and -2x and CCOR (1) in C register bits 61 through 63.

Condition 8: -1x and CCOR (1) in C register bits 61 through 63.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

The partial sum and carry results of the first multiply are looped back in the Multiply Partial Summing network and resummed in the next multiply. This allows the results from the upper Multiply Partial Summing network to be output so that they can be checked.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

Conditions:

C.2 Test the -1x multiplier decodes from C register bits 65 through 67 to the A Product network and upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0006 0000(HEX), in RF0C.
b. Start processor at Micrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test the -1x multiplier decodes from C register bits 63 through 65 to the A Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0018 0000(HEX), in RF0C.
b. Start processor at Micrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test the -1x multiplier decodes from C register bits 61 through 63 to the A Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0060 0000(HEX), in RF0C.
b. Start processor at Micrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.
Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	037F FFFF FFFF FFFF
		C.5 and C.6	00	0DFF FFFF FFFF FFFF
		C.8 and C.9	00	37FF FFFF FFFF FFFF
2	5555 5555 5555 5555	C.2 and C.3	00	03FF FFFF FFFF FFFF
		C.5 and C.6	00	0FFF FFFF FFFF FFFF
		C.8 and C.9	00	3FFF FFFF FFFF FFFF
3	7FFF FFFF FFFF FFFF	C.2 and C.3	00	05FF FFFF FFFF FFFF
		C.5 and C.6	00	17FF FFFF FFFF FFFF
		C.8 and C.9	00	5FFF FFFF FFFF FFFF
4	3333 3333 3333 3333	C.2 and C.3	00	0266 6666 6666 6666
		C.5 and C.6	00	0999 9999 9999 9999
		C.8 and C.9	00	2666 6666 6666 6666
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	0399 9999 9999 9999
		C.5 and C.6	00	0E66 6666 6666 6666
		C.8 and C.9	00	3999 9999 9999 9999

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	00	0400 0000 0000 0000
		C.5 and C.6	00	1000 0000 0000 0000
		C.8 and C.9	00	4000 0000 0000 0000
7	C555 5555 5555 5555	C.2 and C.3	00	02C0 0000 0000 0000
		C.5 and C.6	00	0B00 0000 0000 0000
		C.8 and C.9	00	2C00 0000 0000 0000

Section 2, Subsection 34

This subsection tests -0x decodes in C register bits 67 through 73 to the B Product network.

Hardware Tested:

- Multiplier decoders for C register bits 67 through 73.
- B Product network.
- Multiply Final Adder.

Method:

Testing of the multiplier decoding for -0x in C register bits 67 through 73 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection generate the following decodes to the B Product network and the listed CCOR signals to the Multiply Partial Summing network.

Condition 2: -0x and CCOR (11) in C register bits 71 through 73, and -2x and CCOR (9) in C register bits 69 through 71.

Condition 5: -0x and CCOR (9) in C register bits 69 through 71, -2x and CCOR (7) in C register bits 67 through 69, and 1x in C register bits 71 through 73.

Condition 8: -0x and CCOR (7) in C register bits 67 through 69, and -2x and CCOR (9) in C register bits 69 through 71. Also, 1x in C register bits 65 through 67 to the A Product network.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test -0x multiplier decodes from C register bits 71 through 73 to the B Product network.
a. Write multiplier, 0000 0000 0000 1C00(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

Conditions:

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test -0x multiplier decodes from C register bits 69 through 71 to the B Product network.
a. Write multiplier, 0000 0000 0000 7000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test -0x multiplier decodes from C register bits 67 through 69 to the B Product network.
a. Write multiplier, 0000 0000 0001 C000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
a. Wait halt.
b. Read and report contents of RF0A (MSUM bits 8 through 71).
c. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	01	2AAA AAAA AAAA AAA8
		C.5 and C.6	04	1555 5555 5555 554C
		C.8 and C.9	10	5555 5555 5555 5530
2	5555 5555 5555 5555	C.2 and C.3	01	5555 5555 5555 5554
		C.5 and C.6	04	AAAA AAAA AAAA AAA6
		C.8 and C.9	12	AAAA AAAA AAAA AA98
3	7FFF FFFF FFFF FFFF	C.2 and C.3	01	FFFF FFFF FFFF FFFC
		C.5 and C.6	06	FFFF FFFF FFFF FFF2
		C.8 and C.9	1B	FFFF FFFF FFFF FFC8
4	3333 3333 3333 3333	C.2 and C.3	00	CCCC CCCC CCCC CCCC
		C.5 and C.6	02	CCCC CCCC CCCC CCCA
		C.8 and C.9	0B	3333 3333 3333 3328
5	4CCC CCCC CCCC CCCC	C.2 and C.3	01	3333 3333 3333 3330
		C.5 and C.6	04	3333 3333 3333 3328
		C.8 and C.9	10	CCCC CCCC CCCC CCA0

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	01	5555 5555 5555 5558
		C.5 and C.6	04	AAAA AAAA AAAA AAB4
		C.8 and C.9	12	AAAA AAAA AAAA AAD0
7	C555 5555 5555 5555	C.2 and C.3	00	EAAA AAAA AAAA AAAC
		C.5 and C.6	03	3555 5555 5555 555A
		C.8 and C.9	0C	D555 5555 5555 5568

Section 2, Subsection 35

This subsection tests the -0x decodes in C register bits 67 through 73 to the B Product network and the upper bits of the Multiply Partial Summing network.

Hardware Tested:

- Multiply Partial Summing network upper bits.
- Multiplier decoders for C register bits 67 through 73.
- B Product network.
- Multiply Final Adder.

Method:

Testing of the multiplier decoding for -0x in C register bits 67 through 73 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the B Product network and the listed CCOR signals to the Multiply Partial Summing network.

Condition 2: -0x and CCOR (11) in C register bits 71 through 73, and -2x and CCOR (9) in C register bits 69 through 71.

Condition 5: -0x and CCOR (9) in C register bits 69 through 71, -2x and CCOR (7) in C register bits 67 through 69, and 1x in C register bits 71 through 73.

Condition 8: -0x and CCOR (7) in C register bits 67 through 69, and -2x and CCOR (9) in C register bits 69 through 71. Also, 1x in C register bits 65 through 67 to the A Product network.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

The partial sum and carry results of the first multiply are looped back to the Multiply Partial Summing network and resummed in the next multiply. This allows the results from the upper Multiply Partial Summing network to be output so that they can be checked.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

Conditions:

C.2 Test the -0x multiplier decodes from C register bits 71 through 73 to the B Product network and upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0000 1C00(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test the -0x multiplier decodes from C register bits 69 through 71 to the B Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0000 7000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test the -0x multiplier decodes from C register bits 67 through 69 to the B Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0001 C000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.
Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	0012 AAAA AAAA AAAA
		C.5 and C.6	00	0041 5555 5555 5555
		C.8 and C.9	00	0105 5555 5555 5555
2	5555 5555 5555 5555	C.2 and C.3	00	0015 5555 5555 5555
		C.5 and C.6	00	004A AAAA AAAA AAAA
		C.8 and C.9	00	012A AAAA AAAA AAAA
3	7FFF FFFF FFFF FFFF	C.2 and C.3	00	001F FFFF FFFF FFFF
		C.5 and C.6	00	006F FFFF FFFF FFFF
		C.8 and C.9	00	01BF FFFF FFFF FFFF
4	3333 3333 3333 3333	C.2 and C.3	00	000C CCCC CCCC CCCC
		C.5 and C.6	00	002C CCCC CCCC CCCC
		C.8 and C.9	00	00B3 3333 3333 3333
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	0013 3333 3333 3333
		C.5 and C.6	00	0043 3333 3333 3333
		C.8 and C.9	00	010C CCCC CCCC CCCC

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	00	0015 5555 5555 5555
		C.5 and C.6	00	004A AAAA AAAA AAAA
		C.8 and C.9	00	012A AAAA AAAA AAAA
7	C555 5555 5555 5555	C.2 and C.3	00	000E AAAA AAAA AAAA
		C.5 and C.6	00	0033 5555 5555 5555
		C.8 and C.9	00	00CD 5555 5555 5555

Section 2, Subsection 36

This subsection tests -0x decodes in C register bits 61 through 67 to the A Product network.

Hardware Tested:

- Multiplier decoders for C register bits 61 through 67.
- A Product network.
- Multiply Final Adder.

Method:

Testing of the multiplier decoding for -0x in C register bits 61 through 67 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection generate the following decodes to the B Product network and the listed CCOR signals to the Multiply Partial Summing network.

Condition 2: -0x and CCOR (5) in C register bits 65 through 67, and 1x in C register bits 63 through 65. Also, -2x and CCOR (7) in C register bits 67 through 69 to the B Product network.

Condition 5: -0x and CCOR (3) in C register bits 63 through 65, 1x in C register bits 61 through 63, and -2x and CCOR (5) in C register bits 65 through 67.

Condition 7: -0x and CCOR (1) in C register bits 61 through 63, and -2x and CCOR (3) in C register bits 63 through 65.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

Conditions:

C.2 Test -0x multiplier decodes from C register bits 65 through 67 to the A Product network.
a. Write multiplier, 0000 0000 0007 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test -0x multiplier decodes from C register bits 63 through 65 to the A Product network.
a. Write multiplier, 0000 0000 001C 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test -0x multiplier decodes from C register bits 61 through 63 to the A Product network.
a. Write multiplier, 0000 0000 0070 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.
Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	41	5555 5555 5555 54C0
		C.5 and C.6	05	5555 5555 5555 5300
		C.8 and C.9	6A	AAAA AAAA AAAA AC00
2	5555 5555 5555 5555	C.2 and C.3	4A	AAAA AAAA AAAA AA60
		C.5 and C.6	2A	AAAA AAAA AAAA A980
		C.8 and C.9	55	5555 5555 5555 5600
3	7FFF FFFF FFFF FFFF	C.2 and C.3	6F	FFFF FFFF FFFF FF20
		C.5 and C.6	BF	FFFF FFFF FFFF FC80
		C.8 and C.9	00	0000 0000 0000 0200
4	3333 3333 3333 3333	C.2 and C.3	2C	CCCC CCCC CCCC CCA0
		C.5 and C.6	B3	3333 3333 3333 3280
		C.8 and C.9	99	9999 9999 9999 9A00
5	4CCC CCCC CCCC CCCC	C.2 and C.3	43	3333 3333 3333 3280
		C.5 and C.6	0C	CCCC CCCC CCCC CA00
		C.8 and C.9	66	6666 6666 6666 6800

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	4A	AAAA AAAA AAAA AB40
		C.5 and C.6	2A	AAAA AAAA AAAA AD00
		C.8 and C.9	55	5555 5555 5555 5400
7	C555 5555 5555 5555	C.2 and C.3	33	5555 5555 5555 55A0
		C.5 and C.6	CD	5555 5555 5555 5680
		C.8 and C.9	8A	AAAA AAAA AAAA AA00

Section 2, Subsection 37

This subsection tests the -0x decodes in C register bits 61 through 67 to the A Product network and the upper bits of the Multiply Partial Summing network.

Hardware Tested:

- Multiply Partial Summing network upper bits.
- Multiplier decoders for C register bits 61 through 67.
- A Product network.
- Multiply Final Adder.

Method:

Testing of the multiplier decoding for -0x in C register bits 61 through 67 is done using the hardware controls described in the Section Method.

The multipliers used in this subsection will generate the following decodes to the A Product network and the listed CCOR signals to the Multiply Partial Summing network.

Condition 2: -0x and CCOR (5) in C register bits 65 through 67, and 1x in C register bits 63 through 65. Also, -2x and CCOR (7) in C register bits 67 through 69 to the B Product network.

Condition 5: -0x and CCOR (3) in C register bits 63 through 65, 1x in C register bits 61 through 63, and -2x and CCOR (5) in C register bits 65 through 67.

Condition 7: -0x and CCOR (1) in C register bits 61 through 63, and -2x and CCOR (3) in C register bits 63 through 65.

The negative sign of the last two test patterns will complement all positive or negative controls from all C register decodes to both the A and B Product networks.

The partial sum and carry results of the first multiply are looped back to the Multiply Partial Summing network and resummed in the next multiply. This allows the results from the upper Multiply Partial Summing network to be output so that they can be checked.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

Conditions:

C.2 Test the -0x multiplier decodes from C register bits 65 through 67 to the A Product network and upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0007 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test the -0x multiplier decodes from C register bits 63 through 65 to the A Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 001C 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test the -0x multiplier decodes from C register bits 61 through 63 to the A Product network and the upper Multiply Partial Summing network.
a. Write multiplier, 0000 0000 0070 0000(HEX), in RF0C.
b. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.
Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, and 8 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, and 9 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	0000 0000 0000 0000
		C.8 and C.9	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.2 and C.3	00	0415 5555 5555 5555
		C.5 and C.6	00	1055 5555 5555 5555
		C.8 and C.9	00	4155 5555 5555 5555
2	5555 5555 5555 5555	C.2 and C.3	00	04AA AAAA AAAA AAAA
		C.5 and C.6	00	12AA AAAA AAAA AAAA
		C.8 and C.9	00	4AAA AAAA AAAA AAAA
3	7FFF FFFF FFFF FFFF	C.2 and C.3	00	06FF FFFF FFFF FFFF
		C.5 and C.6	00	1BFF FFFF FFFF FFFF
		C.8 and C.9	00	6FFF FFFF FFFF FFFF
4	3333 3333 3333 3333	C.2 and C.3	00	02CC CCCC CCCC CCCC
		C.5 and C.6	00	0B33 3333 3333 3333
		C.8 and C.9	00	2CCC CCCC CCCC CCCC
5	4CCC CCCC CCCC CCCC	C.2 and C.3	00	0433 3333 3333 3333
		C.5 and C.6	00	10CC CCCC CCCC CCCC
		C.8 and C.9	00	4333 3333 3333 3332

NOTE

The following two patterns are used to check complement control based on B sign.

6	AAAA AAAA AAAA AAAA	C.2 and C.3	00	04AA AAAA AAAA AAAA
		C.5 and C.6	00	12AA AAAA AAAA AAAA
		C.8 and C.9	00	4AAA AAAA AAAA AAAB
7	C555 5555 5555 5555	C.2 and C.3	00	0335 5555 5555 5555
		C.5 and C.6	00	0CD5 5555 5555 5555
		C.8 and C.9	00	3355 5555 5555 5555

Section 2, Subsection 38

This subsection tests C register complement decodes to the A and B Product networks on, C register negative and both B and C registers negative using multiplier values in C register bits 65 through 73.

Hardware Tested:

- Multiplier decoders for C register bits 61 through 73.
- B Product network.
- A Product network.
- Multiply Final Adder.

Method:

The testing of the inversion of the complement signals to B Product network when the C register is negative or no inversion when both the B and C registers are negative is tested using the hardware controls described in the Section Method.

The three multipliers used in this subsection will pull one no complement decode and all the rest of the decodes will be complements to both product networks except when the sign of the multiplicand is negative; then the opposite is sent to the product networks.

All 71 bits of the Full Adder output are checked to ensure that the proper complement decodes were pulled.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Write multiplier, 8000 0000 0000 1000(HEX), in RF0C.
a. Report multiplier; informational.

C.3 Test sign control of complement decodes with decodes in C register bits 69 through 73.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.4 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and reports them in byte 7.
a. Error if not equal to expected result.

Conditions:

C.5 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Write multiplier, 8000 0000 0000 4000(HEX), in RF0C.
a. Report multiplier; informational.

C.7 Test sign control of complement decode with decoder in C register bits 67 through 71.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.8 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.9 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.10 Write multiplier, 8000 0000 0001 0000(HEX), in RF0C.
a. Report multiplier; informational.

C.11 Test sign control of complement decodes with decoder in C register bits 63 through 67.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.12 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.13 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.14 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 3, 7, and 11 report bits 8 through 71 in bits 0 through 63. Conditions 4, 8, and 12 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.3 and C.4	00	0000 0000 0000 0000
		C.7 and C.8	00	0000 0000 0000 0000
		C.11 and C.12	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.3 and C.4	FF	6AAA AAAA AAAA AAAC
		C.7 and C.8	FD	AAAA AAAA AAAA AAB0
		C.11 and C.12	F6	AAAA AAAA AAAA AAC0
2	5555 5555 5555 5555	C.3 and C.4	FF	5555 5555 5555 5556
		C.7 and C.8	FD	5555 5555 5555 5558
		C.11 and C.12	F5	5555 5555 5555 5560
3	7FFF FFFF FFFF FFFF	C.3 and C.4	FF	0000 0000 0000 0002
		C.7 and C.8	FC	0000 0000 0000 0008
		C.11 and C.12	F0	0000 0000 0000 0020
4	3333 3333 3333 3333	C.3 and C.4	FF	9999 9999 9999 999A
		C.7 and C.8	FE	6666 6666 6666 6668
		C.11 and C.12	F9	9999 9999 9999 99A0
5	4CCC CCCC CCCC CCCC	C.3 and C.4	FF	6666 6666 6666 6668
		C.7 and C.8	FD	9999 9999 9999 99A0
		C.11 and C.12	F6	6666 6666 6666 6680

NOTE

The following two patterns are used to check the block of complement inversion when both signs are negative.

6	AAAA AAAA AAAA AAAA	C.3 and C.4	FF	5555 5555 5555 5554
		C.7 and C.8	FD	5555 5555 5555 5550
		C.11 and C.12	F5	5555 5555 5555 5540
7	C555 5555 5555 5555	C.3 and C.4	FF	8AAA AAAA AAAA AAAA
		C.7 and C.8	FE	2AAA AAAA AAAA AAA8
		C.11 and C.12	F8	AAAA AAAA AAAA AAA0

Section 2, Subsection 39

This subsection tests C register complement decodes to the A and B Product networks on C register negative and both B and C registers negative using multiplier values in C register bits 61 through 67.

Hardware Tested:

- Multiplier decoders for C register bits 61 through 73.
- B Product network.
- A Product network.
- Multiply Final Adder.

Method:

The testing of the inversion of the complement signals to B Product network when the C register is negative or no inversion when both the B and C registers are negative is tested using the hardware controls described in the Section Method.

The three multipliers used in this subsection will pull one no complement decode. All the rest of the decodes will be complements to both Product networks except when the sign of the multiplicand is negative; then the opposite is sent to the Product networks.

All 71 bits of the Full Adder output are checked to ensure that the proper complement decodes were pulled.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Write multiplier, 8000 0000 0004 0000(HEX), in RF0C.
a. Report multiplier; informational.

C.3 Test sign control of complement decodes with decodes in C register bits 63 through 67.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.4 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and reports them in byte 7.
a. Error if not equal to expected result.

Conditions:

C.5 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Write multiplier, 8000 0000 0010 0000(HEX), in RF0C.
a. Report multiplier; informational.

C.7 Test sign control of complement decode with decoder in C register bits 61 through 65.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.8 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.9 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.10 Write multiplier, 8000 0000 0040 0000(HEX), in RF0C.
a. Report multiplier; informational.

C.11 Test sign control of complement decodes with decoder in C register bits 61 through 63.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.12 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.13 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.14 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 3, 7, and 11 report bits 8 through 71 in bits 0 through 63. Conditions 4, 8, and 12 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	C.3 and C.4	00	0000 0000 0000 0000
		C.7 and C.8	00	0000 0000 0000 0000
		C.11 and C.12	00	0000 0000 0000 0000
1	4AAA AAAA AAAA AAAA	C.3 and C.4	DA	AAAA AAAA AAAA AB00
		C.7 and C.8	6A	AAAA AAAA AAAA AC00
		C.11 and C.12	55	5555 5555 5555 5000
2	5555 5555 5555 5555	C.3 and C.4	D5	5555 5555 5555 5580
		C.7 and C.8	55	5555 5555 5555 5800
		C.11 and C.12	AA	AAAA AAAA AAAA A800
3	7FFF FFFF FFFF FFFF	C.3 and C.4	C0	0000 0000 0000 0080
		C.7 and C.8	00	0000 0000 0000 0200
		C.11 and C.12	FF	FFFF FFFF FFFF F800
4	3333 3333 3333 3333	C.3 and C.4	E6	6666 6666 6666 6680
		C.7 and C.8	99	9999 9999 9999 9A00
		C.11 and C.12	99	9999 9999 9999 9800
5	4CCC CCCC CCCC CCCC	C.3 and C.4	D9	9999 9999 9999 9A00
		C.7 and C.8	66	6666 6666 6666 6800
		C.11 and C.12	66	6666 6666 6666 6000

NOTE

The following two patterns are used to check the block of complement inversion when both signs are negative.

6	AAAA AAAA AAAA AAAA	C.3 and C.4	D5	5555 5555 5555 5500
		C.7 and C.8	55	5555 5555 5555 5400
		C.11 and C.12	AA	AAAA AAAA AAAA B000
7	C555 5555 5555 5555	C.3 and C.4	E2	AAAA AAAA AAAA AA80
		C.7 and C.8	8A	AAAA AAAA AAAA AA00
		C.11 and C.12	D5	5555 5555 5555 5800

SECTION 3 - CHECK MULTIPLY FINAL ADDER LOGIC

This section verifies the operation of the Multiply Final Adder logic.

Subsection 0 - Test Multiply Final Adder internal array logic for enable detection in the bit 7 position.

Subsection 1 - Test Multiply Final Adder internal array logic for enable detection in the bit 6 position.

Subsection 2 - Test Multiply Final Adder internal array logic for enable detection in the bits 5 through 0 positions.

Subsection 3 - Test Multiply Final Adder internal array Generate/Satisfy logic.

Subsection 4 - Test Multiply Final Adder internal array Generate/Enable/Satisfy logic.

Subsection 5 - Test Multiply Final Adder Carry Interchange logic.

Subsection 6 - Test Multiply Final Adder logic for carry out of 12-bit iteration.

Method:

General

All testing in this section uses the features of the hardware design that do a 12-bit by 64-bit multiply each minor cycle and provide control of each minor cycle by the ALN soft control memory. These controls are used to do one of two types of multiplies: a straight 12-by-64 bit multiply with the results taken directly as a true result from the Final Adder; or as a 24-by-64 bit, 2-iteration, 12-by-64, 12-by-64 multiply with the partial sums and partial carries of the first iteration summed with the partial sums and partial carries of the second iteration with the result taken from the Final Adder.

Hardware Controls

Major Cycle Control: Both microand sequences used in this section use a similar procedure and only vary in the minor cycle control that they cause to be executed in the Multiply/Divide unit. The first microand loads the B register with 64 bits from RF0B and the C register with 64 bits from RF0C. The sequence copies the Multiply/Divide Output mux through the ALN General network and writes it into RF0A. The sequence then copies the Multiply/Divide Output mux through the ALN General network and writes it in RF09. The basic differance between the two microand sequences used is that an extra microand is needed when enabling the partial summary hardware at an earlier minor cycle time to allow for propper data latching.

Minor cycle control: Two types of minor cycle control are used in this section.

a. Nonfeedback multiply (ALN soft control entry points 0, 1 and 2) causes the following sequence of operations in the multiply logic: 1. The C register is enabled but the corresponding enable for clocking of the Multiply Partial Summing network is not set. This is done to overcome the fact that the C register adds 2 bits below the least significant bits of the multiplier and skews the multiplier decodes. The bit not being set effectively discards the results of the multiply by the lower 10 bits of the multiplier. 2. Parity checking is enabled, the C register shifted and the clocking of the Multiply Partial Summing network enabled. 3. The Multiply/Divide Output mux is switched to select MSUM bits 8 through 71 and the minor cycle controls are held in this condition so that the second microand can read this value. 4. When the third microand is executed the minor cycle controls are changed to cause the Multiply/Divide Output mux to select the MSUM bits 0 through 7.

b. Feedback multiply (ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX)) causes the following sequence of operations in the multiply logic: 1. The C register is enabled but the corresponding enable for clocking of the Multiply Partial Summing network is not set. 2. Parity checking is enabled, the C Register is shifted, the checking is enabled, the C register shifted and the clocking of the Multiply Partial Summing network enabled, and the Multiply Partial Summing network feedback is also enabled. 3. The Multiply/Divide Output mux is switched to select the MSUM bits 8 through 71 and the minor cycle controls are held in this condition so that the third microand can read this value. 4. When the fourth microand is executed, the minor cycle controls are changed to cause the Multiply/Divide Output mux to select the MSUM bits 0 through 7.

The following definitions apply to Generate, Enable and Satisfy as used in this document and as seen on the input pins to the Multiply Final Adder.

0 plus 0 (binary) is a Generate.
1 plus 0 (binary) is an Enable.
0 plus 1 (binary) is an Enable.
1 plus 1 (binary) is a Satisfy.

Status Reporting:

All conditions in this section that read and report status will read and check all status listed in under Section Status Format.

Microrands Used:

Sequence Number	Description
10	Run multiply from ALN soft control entry points 0, 1, and 2. This sequence loads the Multiply B register from RF0B, loads the C register from RF0C and writes the Multiply/Divide Output mux in RF0A and RF09.
67	Run multiply from ALN soft control entry points 10, 11, 12, and 13(HEX). This sequence loads the Multiply B register from RF0B, loads the C register from RF0C and writes the Multiply/Divide Output mux in RF0A and RF09.

Initialization:

- Initialization is performed as part of Subsection 0 PP code.
- Write the following ALN soft control memory map (hexadecimal) into the locations specified.

Location	Value to be Written
0	4108 4108
1	D308 0000
2	0108 0000
3	0000 0000
4	0114 D308
5	0000 D908
6	0000 0108
7	0000 0000
8	010C 0114
9	0000 0000
A	0000 0000
B	0000 0000
C	0000 010C
D	0000 0000
E	0000 0000
F	0000 0000

Section Status Format:

The following bit assignments are used by all conditions that report status.

NOTE

Any status not read by a condition and all unused bit positions will be reported as zero. Refer to Section Method for status that will be read and reported.

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
0	Soft Control Register Parity Error byte 0	PFSALN(62)	PFS88/14
1	Soft Control Register Parity Error byte 1	PFSALN(63)	PFS88/15
2	Not Used		
3	Not Used		
4	Not Used		
5	Not Used		
6	Multiply Carry Miscompare byte 0	PFSALN(44)	PFS88/60
7	Multiply Carry Miscompare byte 1	PFSALN(45)	PFS88/61
8	Not Used		
9	Multiply Carry Miscompare byte 2	PFSALN(46)	PFS89/0
10	Multiply Carry Miscompare byte 3	PFSALN(47)	PFS89/1
11	Multiply Carry Miscompare byte 4	PFSALN(48)	PFS89/2
12	Multiply Carry Miscompare byte 5	PFSALN(49)	PFS89/3
13	Multiply Carry Miscompare byte 6	PFSALN(50)	PFS89/4
14	Multiply Carry Miscompare byte 7	PFSALN(51)	PFS89/5
15	Multiply Carry Miscompare byte 8	PFSALN(52)	PFS89/6
16	Multiply C Register Parity Error byte 0	PFSALN(0)	PFS88/16
17	Multiply C Register Parity Error byte 1	PFSALN(1)	PFS88/17
18	Multiply C Register Parity Error byte 2	PFSALN(2)	PFS88/18
19	Multiply C Register Parity Error byte 3	PFSALN(3)	PFS88/19
20	Multiply C Register Parity Error byte 4	PFSALN(4)	PFS88/20
21	Multiply C Register Parity Error byte 5	PFSALN(5)	PFS88/21
22	Multiply C Register Parity Error byte 6	PFSALN(6)	PFS88/22

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
23	Multiply C Register Parity Error byte 7	PFSALN(7)	PFS88/23
24	Multiply B Register Parity Error byte 0	PFSALN(8)	PFS88/24
25	Multiply B Register Parity Error byte 1	PFSALN(9)	PFS88/25
26	Multiply B Register Parity Error byte 2	PFSALN(10)	PFS88/26
27	Multiply B Register Parity Error byte 3	PFSALN(11)	PFS88/27
28	Multiply B Register Parity Error byte 4	PFSALN(12)	PFS88/28
29	Multiply B Register Parity Error byte 5	PFSALN(13)	PFS88/29
30	Multiply B Register Parity Error byte 6	PFSALN(14)	PFS88/30
31	Multiply B Register Parity Error byte 7	PFSALN(15)	PFS88/31
32	DAI Data Parity Error byte 0	PFSOPI(27)	PFS84/48
33	DAI Data Parity Error byte 1	PFSOPI(28)	PFS84/49
34	DAI Data Parity Error byte 2	PFSOPI(29)	PFS84/50
35	DAI Data Parity Error byte 3	PFSOPI(30)	PFS84/51
36	DAI Data Parity Error byte 4	PFSOPI(31)	PFS84/52
37	DAI Data Parity Error byte 5	PFSOPI(32)	PFS84/53
38	DAI Data Parity Error byte 6	PFSOPI(33)	PFS84/54
39	DAI Data Parity Error byte 7	PFSOPI(34)	PFS84/55
40	ALN Micrand Parity Error byte 0	PFSALN(54)	PFS89/8
41	ALN Micrand Parity Error byte 1	PFSALN(55)	PFS89/9
42	ALN Micrand Parity Error byte 2	PFSALN(56)	PFS89/10
43	ALN Micrand Parity Error byte 3	PFSALN(57)	PFS89/11
44	ALN Micrand Parity Error byte 4	PFSALN(58)	PFS89/12
45	ALN Micrand Parity Error byte 5	PFSALN(59)	PFS89/13
46	ALN Micrand Parity Error byte 6	PFSALN(60)	PFS89/14
47	ALN Micrand Parity Error byte 7	PFSALN(61)	PFS89/15
48	Not Used		
49	Not Used		
50	Not Used		
51	Not Used		
52	Processor Did Not Halt Flag from PP Program		
53	Control Store Address bit 0		CSA/53
54	Control Store Address bit 1		CSA/54
55	Control Store Address bit 2		CSA/55

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
56	Control Store Address bit 3		CSA/56
57	Control Store Address bit 4		CSA/57
58	Control Store Address bit 5		CSA/58
59	Control Store Address bit 6		CSA/59
60	Control Store Address bit 7		CSA/60
61	Control Store Address bit 8		CSA/61
62	Control Store Address bit 9		CSA/62
63	Control Store Address bit 10		CSA/63

Section 3, Subsection 0

This subsection tests internal Multiply Final Adder array logic for enable detection in bit 7 position of the array.

Hardware Tested:

- Multiply Final Adder.

Method:

The testing of the Multiply Final Adder logic for internal enable detection will be done using the hardware controls described in the Section Method.

The test patterns used in this subsection start with the Adder array that provides MSUM bits 56 through 63 and walk through the array that provides MSUM bits 16 through 23.

Initialization:

- Write multiplier of 0000 0000 0002 0800(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 If pattern number 6, write negative multiplier 8000 0000 0002 0800(HEX) in RF0C; otherwise, do not change multiplier written by initialization.
a. Report multiplier; informational.

Conditions:

C.3 Test enable detection in bit 7 positions of Final Adder arrays.
- a. Start processor at Micrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
- b. Wait halt.
- c. Read and report contents of RF0A (MSUM bits 8 through 71).
- d. Error if not equal to expected result.

C.4 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
- a. Error if not equal to expected result.

C.5 Read/format and report status according to Section Status Format.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Report First Failure parity information.
- a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 3 reports bits 8 through 71 in bits 0 through 63. Condition 4 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 C0F9	00	0000 0000 0030 FF39
1	0000 0000 00C0 F900	00	0000 0000 30FF 3900
2	0000 0000 C0F9 0000	00	0000 0030 FF39 0000
3	0000 00C0 F900 0000	00	0000 30FF 3900 0000
4	0000 C0F9 0000 0000	00	0030 00FF 3900 0000
5	00C0 F900 0000 0000	00	30FF 3900 0000 0000
6	C0F9 0000 0000 0000	EF	FF39 0000 0000 0000

Section 3, Subsection 1

This subsection tests internal Multiply Final Adder array logic for enable detection in bit 6 position of the array.

Hardware Tested:

- Multiply Final Adder.

Method:

The testing of the Multiply Final Adder logic for internal enable detection will be done using the hardware controls described in the Section Method.

The test patterns used in this subsection start with the array that provides MSUM bits 48 through 55 and walks through the array that provides MSUM bits 16 through 23.

Initialization:

- Write multiplier of 0000 0000 000A 2800(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
- a. Report test pattern; informational.

C.2 If pattern number 5, write negative multiplier 8000 0000 0002 2800(HEX) in RF0C; otherwise do not change multiplier written by initialization.
- a. Report multiplier; informational.

C.3 Test enable detection in bit 6 positions of Final Adder arrays.
- a. Start processor at Micrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
- b. Wait halt.
- c. Read and report contents of RF0A (MSUM bits 8 through 71).
- d. Error if not equal to expected result.

C.4 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
- a. Error if not equal to expected result.

C.5 Read/format and report status according to Section Status Format.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

Conditions:

C.6 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.3.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 3 reports bits 8 through 71 in bits 0 through 63. Condition 4 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 00E4 6DA0	00	0000 0001 21FF 2C20
1	0000 0000 E46D A000	00	0000 0121 FF2C 2000
2	0000 00E4 6DA0 0000	00	0001 21FF 2C20 0000
3	0000 E46D A000 0000	00	0121 FF2C 2000 0000
4	00E4 6DA0 0000 0000	01	21FF 2C20 0000 0000
5	E46D A000 0000 0000	DC	FF2C 2000 0000 0000

Section 3, Subsection 2

This subsection tests Multiply Final Adder internal array logic for enable detection in the bits 5 through 0 positions of the array.

Hardware Tested:

- Multiply Final Adder.

Method:

The testing of the Multiply Final Adder internal array enable detection logic is to be done using the hardware controls described in the Section Method. The test patterns start with bit 5 enable detection in the array that provides MSUM bits 48 through 55. The subsection tests bit 5 position in each adder array until it reaches the array that provides MSUM bits 8 through 15. Then the subsection tests bit 4 for the same arrays, then bit 3, bit 2, and bit 0.

The byte value of FF(HEX) in the expected result corresponds to the Multiply Final Adder array that is being tested by that test pattern.

Initialization:

- Write multiplier, 0000 0000 0028 9C00(HEX), in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test Multiply Final Adder internal enable detection.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 in bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0013 7ED1	00	0000 0000 62FF FD54
1	0000 0000 137E D100	00	0000 0062 FFFD 5400
2	0000 0013 7ED1 0000	00	0000 62FF FD54 0000
3	0000 137E D100 0000	00	0062 FFFD 5400 0000
4	0013 7ED1 0000 0000	00	62FF FD54 0000 0000
5	137E D100 0000 0000	62	FFFD 5400 0000 0000
6	0000 0000 0026 FDA2	00	0000 0000 C5FF FAA8
7	0000 0000 26FD A200	00	0000 00C5 FFFA A800
8	0000 0026 FDA2 0000	00	0000 C5FF FAA8 0000
9	0000 26FD A200 0000	00	00C5 FFFA A800 0000
A	0026 FDA2 0000 0000	00	C5FF FAA8 0000 0000
B	26FD A200 0000 0000	C5	FFFA A800 0000 0000
C	0000 0000 004D FB44	00	0000 0001 8BFF F550
D	0000 0000 4DFB 4400	00	0000 018B FFF5 5000
E	0000 004D FB44 0000	00	0001 8BFF F550 0000
F	0000 4DFB 4400 0000	00	018B FFF5 5000 0000
10	004D FB44 0000 0000	01	8BFF F550 0000 0000
11	4DFB 4400 0000 0000	8B	FFF5 5000 0000 0000
12	0000 0000 001B F688	00	0000 0000 8DFF EAA0
13	0000 0000 1BF6 8800	00	0000 008D FFEA A000

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
14	0000 001B F688 0000	00	0000 8DFF EAA0 0000
15	0000 1BF6 8800 0000	00	008D FFEA A000 0000
16	001B F688 0000 0000	00	8DFF EAA0 0000 0000
17	1BF6 8800 0000 0000	8D	FFEA A000 0000 0000
18	0000 0000 0037 ED10	00	0000 0001 1BFF D540
19	0000 0000 37ED 1000	00	0000 011B FFD5 4000
1A	0000 0037 ED10 0000	00	0001 1BFF D540 0000
1B	0000 37ED 1000 0000	00	011B FFD5 4000 0000
1C	0037 ED10 0000 0000	01	1BFF D540 0000 0000
1D	37ED 1000 0000 0000	1B	FFD5 4000 0000 0000
1E	0000 0000 006F DA20	00	0000 0002 37FF AA80
1F	0000 0000 6FDA 2000	00	0000 0237 FFAA 8000
20	0000 006F DA20 0000	00	0002 37FF AA80 0000
21	0000 6FDA 2000 0000	00	0237 FFAA 8000 0000
22	006F DA20 0000 0000	02	37FF AA80 0000 0000
23	6FDA 2000 0000 0000	37	FFAA 8000 0000 0000

Section 3, Subsection 3

This subsection tests the Multiply Final Adder internal array Generate/Satisfy logic.

Hardware Tested:

- Multiply Final Adder.

Method:

The testing of the Multiply Final Adder internal Generate/Satisfy logic uses a single Generate with a Satisfy in the next higher bit position. The hardware controls described in the Section Method will be used.

The multiplier and the test patterns used in this subsection will cause a single Generate in 1 bit position with the Satisfy in the bit position above it. The test patterns will start with the result in MSUM bit 65 and walk to the result in MSUM bit 3.

Expected Results:

The expected result in MSUM bits 0 through 71 is the test pattern left-shifted by six positions.

Example for iteration count 3D(HEX):

Test pattern equals 2000 0000 0000 0000(HEX).
Expected result in MSUM bits 0 through 71 (RF09 and RF0A) equals 08 0000 0000 0000 0000(HEX).

Initialization:

- Write multiplier, 0000 0000 0001 FC00(HEX), in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test Multiply Final Adder internal Generate/Satisfy logic.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to the test pattern left-shifted six positions.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to the upper 6 bits of the test pattern.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns (Hexadecimal):

Iteration Count	Test Pattern
0	0000 0000 0000 0001
1	0000 0000 0000 0002
2	0000 0000 0000 0004
3	0000 0000 0000 0008
4	0000 0000 0000 0010
.	.
.	.
.	.
3B	0800 0000 0000 0000
3C	1000 0000 0000 0000
3D	2000 0000 0000 0000
3E	4000 0000 0000 0000

Section 3, Subsection 4

This subsection tests Multiply Final Adder internal array Generate/Enable/Satisfy logic.

Hardware Tested:

- Multiply Final Adder.

Method:

The testing of the Multiply Final Adder Generate/Enable/Satisfy internal array logic is done by walking a Satisfy away from a Generate. The hardware controls are described in the Section Method.

Test patterns 0 through 1B(HEX) will test the adder arrays that provide MSUM bytes 1, 3, 5, and 7. Test patterns 1C(HEX) through 38(HEX) will test the adder arrays that provide MSUM bytes 0, 2, 4, and 6.

Expected Results:

The expected results are always the input test pattern left-shifted six places in MSUM.

Example for pattern number 2F(HEX):

Test pattern equals 6000 E000 E000 E000(HEX).
Expected result in MSUM bits 0 through 71 (RF09 and RF0A) equals 18 0038 0038 0038 0038(HEX).

Initialization:

- Write multiplier, 0000 0000 0001 FC00(HEX), in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test Multiply Final Adder internal array Generate/Enable/Satisfy logic.
a. Start processor at Microrand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to test pattern left-shifted six positions.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to upper 6 bits of the test pattern.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns (Hexadecimal):

Iteration Count	Test Pattern
0	000C 000C 000C 000C
1	001C 001C 001C 001C
2	003C 003C 003C 003C
3	007C 007C 007C 007C
4	00FC 00FC 00FC 00FC
5	01FC 01FC 01FC 01FC
6	03FC 03FC 03FC 03FC
7	0018 0018 0018 0018
8	0038 0038 0038 0038
9	0078 0078 0078 0078
A	00F8 00F8 00F8 00F8
B	01F8 01F8 01F8 01F8
C	03F8 03F8 03F8 03F8
D	0030 0030 0030 0030
E	0070 0070 0070 0070
F	00F0 00F0 00F0 00F0
10	01F0 01F0 01F0 01F0
11	03F0 03F0 03F0 03F0
12	0060 0060 0060 0060
13	00E0 00E0 00E0 00E0
14	01E0 01E0 01E0 01E0
15	03E0 03E0 03E0 03E0
16	00C0 00C0 00C0 00C0
17	01C0 01C0 01C0 01C0
18	03C0 03C0 03C0 03C0

Iteration Count	Test Pattern
19	0180 0180 0180 0180
1A	0380 0380 0380 0380
1B	0300 0300 0300 0300
1C	0C00 0C00 0C00 0C00
1D	1C00 1C00 1C00 1C00
1E	3C00 3C00 3C00 3C00
1F	7C00 7C00 7C00 7C00
20	7C00 FC00 FC00 FC00
21	7C01 FC01 FC01 FC00
22	7C03 FC03 FC03 FC00
23	1800 1800 1800 1800
24	3800 3800 3800 3800
25	7800 7800 7800 7800
26	7800 F800 F800 F800
27	7801 F801 F801 F800
28	7803 F803 F803 F803
29	3000 3000 3000 3000
2A	7000 7000 7000 7000
2B	7000 F000 F000 F000
2C	7001 F001 F001 F000
2D	7003 F003 F003 F000
2E	6000 6000 6000 6000
2F	6000 E000 E000 E000
30	6001 E001 E001 E000
31	6003 E003 E003 E000
32	4000 C000 C000 C000

Iteration Count	Test Pattern
33	4001 C001 C001 C000
34	4003 C003 C003 C000
35	0001 8001 8001 8000
36	0003 8003 8003 8000
38	0003 0003 0003 0003

Section 3, Subsection 5

This subsection tests the Multiply Final Adder Carry Interchange logic.

Hardware Tested:

- Multiply Final Adder Carry Interchange.

Method:

The testing of the Multiply Final Adder Carry Interchange for handling of Generates/Enables/ Satisfies is done by walking a Satisfy away from a Generate. The hardware controls are described in the Section Method.

The multiplier and the test pattern used should cause a Generate from the lowest order bit position set in the test pattern to propagate via the Carry Interchange to the highest order bit position set in the test pattern.

Expected Results:

The expected results are always the input test pattern left-shifted six places in MSUM bits 0 through 71.

Example for Test Pattern 6:

Test pattern equals 7FFF FFFF FFFF FFFC(HEX).
Expected result in MSUM bits 0 through 71 (RF09 and RF0A) equals 1F FFFF FFFF FFFF FF00(HEX).

Initialization:

- Write multiplier, 0000 0000 0001 FC00(HEX), in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test Multiply Final Adder Carry Interchange logic.
a. Start processor at Microand Sequence 10; run multiply from ALN soft control entries 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to the test pattern left-shifted six positions.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to the upper 8 bits of the test pattern.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns (Hexadecimal):

Iteration Count	Test Pattern
0	0000 0000 0000 FFFC
1	0000 0000 00FF FFFC
2	0000 0000 FFFF FFFC
3	0000 00FF FFFF FFFC
4	0000 FFFF FFFF FFFC
5	00FF FFFF FFFF FFFC
6	7FFF FFFF FFFF FFFC
7	0000 0000 00FF FC00
8	0000 0000 FFFF FC00
9	0000 00FF FFFF FC00

Iteration Count	Test Pattern
A	0000 FFFF FFFF FC00
B	00FF FFFF FFFF FC00
C	7FFF FFFF FFFF FC00
D	0000 0000 FFFC 0000
E	0000 00FF FFFC 0000
F	0000 FFFF FFFC 0000
10	00FF FFFF FFFC 0000
11	7FFF FFFF FFFC 0000
12	0000 00FF FC00 0000
13	0000 FFFF FC00 0000
14	00FF FFFF FC00 0000
15	7FFF FFFF FC00 0000
16	0000 FFFC 0000 0000
17	00FF FFFC 0000 0000
18	7FFF FFFC 0000 0000
19	00FF FC00 0000 0000
1A	7FFF FC00 0000 0000
1B	7FFC 0000 0000 0000

Section 3, Subsection 6

This subsection tests the Multiply Final Adder logic for carry out of a 12-bit iteration.

Hardware Tested:

- Multiply Final Adder Enable/Generate determination for PCR2 bits 65 through 68 and PSR2 bits 64 through 67.
- Multiply Final Adder logic for carry out of a 12-bit iteration.

Method:

The testing of the Multiply Final Adder for carry out of a 12-bit iteration will be performed using the hardware controls described in the Section Method.

The multiplier and the test pattern will be used to test the 12-bit iteration carry logic. This carry will be looped back into the lowest bit of the Multiply Partial Summing network and summed with the remaining bits in PCR2/PSR2 and then output as MSUM bits 0 through 71 through the Adder.

Initialization:

- Write multiplier, FFFF FFFF FFFF F800(HEX), in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test Multiply Final Adder Carry out of 12-bit Iteration logic.
a. Start processor at Microrand Sequence 67; run multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to 0000 0000 0000 0000(HEX).

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to 0000 0000 0000 0000(HEX).

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns (Hexadecimal):

Iteration Count	Test Pattern
0	FFFF FFFF FFFF FFFE
1	FFFF FFFF FFFF FFFD
2	FFFF FFFF FFFF FFFB
3	FFFF FFFF FFFF FFF7
4	FFFF FFFF FFFF FFEF
5	FFFF FFFF FFFF FFDF
6	FFFF FFFF FFFF FFBF
7	FFFF FFFF FFFF FF7F
8	FFFF FFFF FFFF FEFF
9	FFFF FFFF FFFF FDFF
A	FFFF FFFF FFFF FBFF
B	FFFF FFFF FFFF F9FF
C	FFFF FFFF FFFF F8FF
D	FFFF FFFF FFFF F87F

SECTION 4 - CHECK B REGISTER ENTRY CONTROLS

This section verifies the operation of the B register, its entry controls, unsigned entry and unsigned multiply.

Subsection 0 - Test B register 32-bit sign selection and extension.

Subsection 1 - Test B register 48-bit entry, sign extended with bit 0 as sign.

Subsection 2 - Test Signed/Unsigned Multiply B register entry controls. The A and B Product networks for disable of B register bit 0 fanout and disable of B and C sign to the Product network decoders.

Subsection 3 - Test B register entry of MSUM bits 8 through 71.

Method:

All subsections in this section use the same ALN soft control memory locations to condition the multiply for a 12-by-64 multiply.

The basic operation of these controls is as follows:

Enter multiplier into C register, shift C register and discard the results of multiply by the lower 10 bits of the C register. Multiply 12 bits by 64 bits and clock Multiply Partial Summing network to save this result. Change Output mux controls to select MSUM bits 8 through 71 and freeze multiply unit until entry point 2 or 12(HEX) is executed. Entry points 2 or 12(HEX) will change the Output mux selection to MSUM bits 0 through 7 to Output mux bits 48 through 55 and again freeze the multiply unit.

The only difference between the sequence executed in entry points 0, 1, and 2 and that in 10(HEX), 11(HEX), and 12(HEX) is that entry point 10(HEX) has V8 set in it to cause the C register to be loaded without sign extension into bits 0 through 7.

All conditions in this section that report status will read/format and report all status listed in the Section Status Format.

Initialization:

Initialization is performed as part of Subsection 0 PP code.

- Write the following ALN soft control memory map (hexadecimal) into the locations listed.

Location	32-bit Value to be Written
0	4108 4188
1	D308 D308
2	0108 0108
3	0000 0000
4	0114 0114
5	0000 0000
6	0000 0000
7	0000 0000
8	010C 010C
9	0000 0000
A	0000 0000
B	0000 0000

Section Status Format:

The following bit assignments are used by all conditions that report status.

NOTE

Any status not read by a condition and all unused bit positions will be reported as zero. See Section Method for status that will be read and reported.

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
0	Soft Control Register Parity Error byte 0	PFSALN(62)	PFS88/14
1	Soft Control Register Parity Error byte 1	PFSALN(63)	PFS88/15
2	Not Used		
3	Not Used		
4	Not Used		
5	Not Used		
6	Multiply Carry Miscompare byte 0	PFSALN(44)	PFS88/60
7	Multiply Carry Miscompare byte 1	PFSALN(45)	PFS88/61
8	Not Used		
9	Multiply Carry Miscompare byte 2	PFSALN(46)	PFS89/0
10	Multiply Carry Miscompare byte 3	PFSALN(47)	PFS89/1
11	Multiply Carry Miscompare byte 4	PFSALN(48)	PFS89/2

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
12	Multiply Carry Miscompare byte 5	PFSALN(49)	PFS89/3
13	Multiply Carry Miscompare byte 6	PFSALN(50)	PFS89/4
14	Multiply Carry Miscompare byte 7	PFSALN(51)	PFS89/5
15	Multiply Carry Miscompare byte 8	PFSALN(52)	PFS89/6
16	Multiply C Register Parity Error byte 0	PFSALN(0)	PFS88/16
17	Multiply C Register Parity Error byte 1	PFSALN(1)	PFS88/17
18	Multiply C Register Parity Error byte 2	PFSALN(2)	PFS88/18
19	Multiply C Register Parity Error byte 3	PFSALN(3)	PFS88/19
20	Multiply C Register Parity Error byte 4	PFSALN(4)	PFS88/20
21	Multiply C Register Parity Error byte 5	PFSALN(5)	PFS88/21
22	Multiply C Register Parity Error byte 6	PFSALN(6)	PFS88/22
23	Multiply C Register Parity Error byte 7	PFSALN(7)	PFS88/23
24	Multiply B Register Parity Error byte 0	PFSALN(8)	PFS88/24
25	Multiply B Register Parity Error byte 1	PFSALN(9)	PFS88/25
26	Multiply B Register Parity Error byte 2	PFSALN(10)	PFS88/26
27	Multiply B Register Parity Error byte 3	PFSALN(11)	PFS88/27
28	Multiply B Register Parity Error byte 4	PFSALN(12)	PFS88/28
29	Multiply B Register Parity Error byte 5	PFSALN(13)	PFS88/29
30	Multiply B Register Parity Error byte 6	PFSALN(14)	PFS88/30
31	Multiply B Register Parity Error byte 7	PFSALN(15)	PFS88/31
32	DAI Data Parity Error byte 0	PFSOPI(27)	PFS84/48
33	DAI Data Parity Error byte 1	PFSOPI(28)	PFS84/49
34	DAI Data Parity Error byte 2	PFSOPI(29)	PFS84/50
35	DAI Data Parity Error byte 3	PFSOPI(30)	PFS84/51
36	DAI Data Parity Error byte 4	PFSOPI(31)	PFS84/52
37	DAI Data Parity Error byte 5	PFSOPI(32)	PFS84/53
38	DAI Data Parity Error byte 6	PFSOPI(33)	PFS84/54
39	DAI Data Parity Error byte 7	PFSOPI(34)	PFS84/55
40	ALN Micrand Parity Error byte 0	PFSALN(54)	PFS89/8

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
41	ALN Micrand Parity Error byte 1	PFSALN(55)	PFS89/9
42	ALN Micrand Parity Error byte 2	PFSALN(56)	PFS89/10
43	ALN Micrand Parity Error byte 3	PFSALN(57)	PFS89/11
44	ALN Micrand Parity Error byte 4	PFSALN(58)	PFS89/12
45	ALN Micrand Parity Error byte 5	PFSALN(59)	PFS89/13
46	ALN Micrand Parity Error byte 6	PFSALN(60)	PFS89/14
47	ALN Micrand Parity Error byte 7	PFSALN(61)	PFS89/15
48	Not Used		
49	Not Used		
50	Not Used		
51	Not Used		
52	Processor Did Not Halt Flag from PP Program		
53	Control Store Address bit 0		CSA/53
54	Control Store Address bit 1		CSA/54
55	Control Store Address bit 2		CSA/55
56	Control Store Address bit 3		CSA/56
57	Control Store Address bit 4		CSA/57
58	Control Store Address bit 5		CSA/58
59	Control Store Address bit 6		CSA/59
60	Control Store Address bit 7		CSA/60
61	Control Store Address bit 8		CSA/61
62	Control Store Address bit 9		CSA/62
63	Control Store Address bit 10		CSA/63

Section 4, Subsection 0

This subsection tests B register 32-bit sign selection and extension.

Hardware Tested:

- B register input mux.
- B register sign selection logic.
- B register mux controls from functional unit micrand bits 8 through 11.

Method:

The testing of the B register 32-bit sign selection and extension will be done using the ALN soft control described in the Section Method with the micrand sequence listed in this subsection. These controls will enter a value into the B register; 32-bit sign extended. This value will be multiplied in condition 3 by a -1x decode from the C register bits 71 through 73, and in condition 7 by two 1x decodes from C register bits 69 through 73.

The output will be taken as MSUM bits 0 through 71 and checked. The expected result, whose multiplier sign is the same as the selected B register sign bit, reflects the true B register sign extension in MSUM bits 0 through 39, except on a zero result when all zeros are output.

Micrands Used:

Sequence Number	Description
14	Run 32-bit B register Multiply from ALN soft control entry points 0, 1, and 2. This sequence uses B register controls U0 and U3.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
- a. Report test pattern; informational.

C.2 Write multiplier of 8000 0000 0000 0800(HEX) in RF0C.
- a. Report multiplier; informational.

C.3 Run 32-bit multiply.
- a. Start processor at Micrand Sequence 14; run 32-bit B register multiply from ALN soft control entry points 0, 1, and 2.
- b. Wait halt.
- c. Read and report contents of RF0A (MSUM bits 8 through 71).
- d. Error if not equal to expected result.

C.4 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
- a. Error if not equal to expected result.

C.5 Read/format and report status according to Section Status Format.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Write multiplier of 0000 0000 0000 1400(HEX) in RF0C.
- a. Report multiplier; informational.

Conditions:

C.7 Run 32-bit multiply.
- a. Start processor at Micrand Sequence 14; run 32-bit B register multiply from ALN soft control entry points 0, 1, and 2.
- b. Wait halt.
- c. Read and report contents of RF0A (MSUM bits 8 through 71).
- d. Error if not equal to expected result.

C.8 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
- a. Error if not equal to expected result.

C.9 Read/format and report status according to Section Status Format.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.10 Report First Failure parity information.
- a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 3 and 7 report bits 8 through 71 in bits 0 through 63. Conditions 4 and 8 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	8000 0000 0000 0000	C.3 and C.4	00	0000 0000 0000 0000
		C.7 and C.8	00	0000 0000 0000 0000
1	4000 0000 0000 0000	C.3 and C.4	00	0000 0000 0000 0000
		C.7 and C.8	00	0000 0000 0000 0000
2	FFFF FFFF 0000 0000	C.3 and C.4	00	0000 0000 0000 0000
		C.7 and C.8	00	0000 0000 0000 0000
3	0000 0000 8000 0000	C.3 and C.4	FF	FFFF FFFF 8000 0000
		C.7 and C.8	00	0000 0001 8000 0000

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
4	0000 0000 FFFF FFFE	C.3 and C.4	FF	FFFF FFFF FFFF FFFE
		C.7 and C.8	00	0000 0000 0000 0006
5	FFFF FFFF 7FFF FFFE	C.3 and C.4	FF	FFFF FFFF 8000 0002
		C.7 and C.8	00	0000 0001 7FFF FFFA

Section 4, Subsection 1

This subsection tests B register 48-bit entry, zero filled with bit 0 as sign.

Hardware Tested:

- B register input mux.
- B register sign selection logic.
- B register mux controls from functional unit micrand bits U0 through U3.

Method:

The testing of B register 48-bit entry and zero fill will be performed using the ALN soft control described in the Section Method and the micrand sequence listed in this subsection.

These controls will enter a 48-bit value into the B register, zero fill bits 1 through 15, and use bit 0 as sign. This value will be multiplied in condition 3 by a -1x decode from C register bits 71 through 73, and in condition 7 by two 1x decodes from C register bits 69 through 73.

The output will be taken as MSUM bits 0 through 71 and checked. The expected result, whose multiplier sign is the same as the selected B register sign bit, reflects the true B register sign extension in MSUM bits 0 through 23, except for a zero result when all zeros are output.

Micrands Used:

Sequence Number	Description
15	Run 48-bit B register multiply from ALN soft control entry points 0, 1, and 2. This sequence uses B register controls U0, U2, and U3.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Write multiplier of 8000 0000 0000 0800(HEX) in RF0C.
a. Report multiplier; informational.

C.3 Run 48-bit multiply.
a. Start processor at Micrand Sequence 15; run 48-bit B register multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.4 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.5 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Write multiplier of 0000 0000 0000 1400(HEX) in RF0C.
a. Report multiplier; informational.

C.7 Run 48-bit multiply.
a. Start processor at Micrand Sequence 15; run 48-bit B register multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.8 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.9 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.10 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 3 and 7 report bits 8 through 71 in bits 0 through 63. Conditions 4 and 8 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	4000 0000 0000 0000	C.3 and C.4	00	0000 0000 0000 0000
		C.7 and C.8	00	0000 0000 0000 0000
1	7FFF 0000 0000 0000	C.3 and C.4	00	0000 0000 0000 0000
		C.7 and C.8	00	0000 0000 0000 0000
2	7FFF 0000 8000 0000	C.3 and C.4	FF	FFFF FFFF 8000 0000
		C.7 and C.8	00	0000 0001 8000 0000
3	8000 FFFF 7FFF FFFE	C.3 and C.4	FF	FFFF 0000 8000 0002
		C.7 and C.8	00	0002 FFFE 7FFF FFFA
4	8000 0000 0000 0001	C.3 and C.4	FF	FFFF FFFF FFFF FFFF
		C.7 and C.8	00	0000 0000 0000 0002

Section 4, Subsection 2

This subsection tests Signed/Unsigned Multiply B register entry controls, the A and B Product network disable of B register bit 0 fanout and disable of B and C Sign to the Product network decoders.

Hardware Tested:

- B register entry controls from U1 and U13.
- A Product network for B register bit 0 fanout disable when U13 is active.
- B Product network for B register bit 0 fanout disable when U13 is active.
- Not U1 and Not U4 disable of B and C sign selection.

Method:

The testing of the B register entry controls from U1 and U13 and the disable of B register bit 0 fanouts in Product networks A and B will be performed using the ALN soft control described in the Section Method and the micrand sequence listed under Micrands Used.

The operation of each condition that executes a micrand sequence is described next. All conditions read the test pattern from RF0B and write the MSUM bits into RF09 and RF0A.

Condition 2: This condition loads 48 bits of the test pattern into the B register, uses bit 0 as sign, and zero fills bits 1 through 15. This B value is multiplied by a 1x decode in the B Product network. This multiply is signed from the B register but B register bit 0 is not extended.

Condition 5: This condition loads 64 bits into the B register and then does an unsigned multiply by a 1x decode in the B Product network.

Condition 8: This condition loads 64 bits into the B register and does an unsigned multiply by a 1x decode in the A Product network.

Condition 11: This condition loads 64 bits into the B register and does an unsigned multiply by a 1x decode in the B Product network. The multiplier will have bit 0 set. This bit should be disabled to the Product network decoders.

All sequences will have their results taken from MSUM bits 0 through 71.

Micrands Used:

Sequence Number	Description
16	Run Signed/Unsigned B register multiply from soft control entry points 10(HEX), 11(HEX), and 12(HEX). This sequence uses U1 XOR U13 for B register entry.
17	Run Unsigned Multiply from soft control entry points 10 (HEX), 11(HEX), and 12(HEX). This sequence uses NOT U1 XOR U13 for B register entry.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Write multiplier 0000 0000 0000 0800(HEX) in RF0C.
b. Report test pattern; informational.

Conditions:

C.2 Test Signed/Unsigned B register multiply.
a. Start processor at Microrand Sequence 16; run Sign/Unsigned B register multiply from ALN soft control entry points 10(HEX), 11 (HEX), and 12(HEX).
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test Unsigned multiply and B Product network.
a. Start processor at Microrand Sequence 17; run Unsigned multiply from ALN soft control entry points 10(HEX), 11(HEX), and 12(HEX).
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.6 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.7 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Test Unsigned multiply and A Product network.
a. Write multiplier, 0000 0000 0002 000(HEX), in RF0C.
b. Start processor at Microrand Sequence 17; run Unsigned multiply from ALN soft control entry points 10(HEX), 11(HEX), and 12(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

C.9 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.10 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Test Unsigned multiply with negative multiplier.
a. Write multiplier 8000 0000 0000 0800 in RF0C.
b. Start processor at Microrand Sequence 17; run Unsigned multiply from ALN soft control entry points 10(HEX), 11(HEX), and 12(HEX).
c. Wait halt.
d. Read and report contents of RF0A (MSUM bits 8 through 71).
e. Error if not equal to expected result.

Conditions:

C.12 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.13 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.14 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.1.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Conditions 2, 5, 8, and 11 report bits 8 through 71 in bits 0 through 63. Conditions 3, 6, 9, and 12 report bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	7FFF 0000 0000 0000	C.2 and C.3	00	0000 0000 0000 0000
		C.5 and C.6	00	7FFF 0000 0000 0000
		C.8 and C.9	1F	FFC0 0000 0000 0000
		C.11 and C.12	00	7FFF 0000 0000 0000
1	AAAA AAAA AAAA AAAA	C.2 and C.3	FF	0000 5555 5555 5556
		C.5 and C.6	00	AAAA AAAA AAAA AAAA
		C.8 and C.9	2A	AAAA AAAA AAAA AA80
		C.11 and C.12	00	AAAA AAAA AAAA AAAA
2	D555 5555 5555 5555	C.2 and C.3	FF	0000 AAAA AAAA AAAB
		C.5 and C.6	00	D555 5555 5555 5555
		C.8 and C.9	35	5555 5555 5555 5540
		C.11 and C.12	00	D555 5555 5555 5555

Section 4, Subsection 3

This subsection tests B register entry of MSUM bits 8 through 71.

Hardware Tested:

- B register input mux.
- B register mux controls from functional unit micrand.
- Disable of B register parity checking when U0 not present.

Method:

Testing of the B register entry of MSUM bits 8 through 71 is done using the ALN soft control described in the Section Method and the following signal flow under micrand control.

The first micrand will cause an unsigned multiply of the test pattern in RF0B by a multiplier from RF0C that will cause the test patterns to be run on the Final Adder as MSUM bits 8 through 71. The second micrand will send a sensitizing pattern to the B register from RF0D. The B register controls should select MSUM bits 8 through 71 (equal to test pattern) and an unsigned multiply is done by a new multiplier from RF08. This multiply should cause the test pattern to appear left-shifted two places in MSUM bits 0 through 71. The third and fourth micrands write MSUM 0 through 71 into the register files.

Micrands Used:

Sequence Number	Description
18	Run loop MSUM bits 8 through 71 to B register from ALN soft control entry points 10(HEX), 11(HEX), and 12(HEX). This sequence does two unsigned multiplies and then outputs the results.

Initialization:

- Write 0000 0000 0000 0800(HEX) in RF0C; first multiplier.
- Write FFFF FFFF 7FFF FFFF(HEX) in RF07; sensitizing pattern for second multiply.
- Write 0000 0000 0000 2000(HEX) in RF08; second multiplier.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test loop MSUM bits 8 through 71.
a. Start processor at Micrand Sequence 18; run Loop MSUM bits 8 through 71 to B register from ALN soft control entry points 10 (HEX), 11(HEX), and 12(HEX).
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 in bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0000	00	0000 0000 0000 0000
1	FFFF FFFF FFFF FFFF	FF	FFFF FFFF FFFF FFFC
2	3333 3333 3333 3333	00	CCCC CCCC CCCC CCCC
3	CCCC CCCC CCCC CCCC	FF	3333 3333 3333 3330
4	5555 5555 5555 5555	01	5555 5555 5555 5554
5	AAAA AAAA AAAA AAAA	FE	AAAA AAAA AAAA AAA8
6	DCDC DCDC DCDC DCDC	FF	7373 7373 7373 7370

SECTION 5 - CHECK C REGISTER ENTRY AND 170 MULTIPLY CONTROLS

This section verifies the following: operation of the lower C register, 170 mode control of the B and C register signs and 170 mode control of the multiplier decoders.

Subsection 0 - Test C register for copies of C register bits from ADATC.

Subsection 1 - Test C register entry of sign under 170 mode control.

Subsection 2 - Test B register sign selection and entry under 170 mode control.

Subsection 3 - Test Complement Correction signals for proper operation under 170 mode control.

Method:

Refer to Subsection Method for detailed description.

All conditions in this section that report status will report all status listed in the Section Status Format.

Initialization:

Initialization is performed as part of Subsection 0 PP code.

- Write the following ALN soft control memory map (hexadecimal) into the locations listed.

Location	32-bit Value to be Written
0	5108 4108
1	C308 D308
2	0108 0108
3	0000 0000
4	0114 0114
5	0000 0000
6	0000 0000
7	0000 0000
8	010C 0000
9	0000 0000
A	0000 0000
B	0000 0000

- Write multiplicand for Subsections 0 and 1, 0000 7ABC 7ABC 0000(HEX), in RF0B.

Section Status Format:

The following bit assignments are used by all Conditions that Report Status.

NOTE

Any status not read by a condition and all unused bit positions will be reported as zero. See Section Method for status that will be read and reported.

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
0	Soft Control Register Parity Error byte 0	PFSALN(62)	PFS88/14
1	Soft Control Register Parity Error byte 1	PFSALN(63)	PFS88/15
2	Not Used		
3	Not Used		
4	Not Used		
5	Not Used		
6	Multiply Carry Miscompare byte 0	PFSALN(44)	PFS88/60
7	Multiply Carry Miscompare byte 1	PFSALN(45)	PFS88/61
8	Not Used		
9	Multiply Carry Miscompare byte 2	PFSALN(46)	PFS89/0
10	Multiply Carry Miscompare byte 3	PFSALN(47)	PFS89/1
11	Multiply Carry Miscompare byte 4	PFSALN(48)	PFS89/2
12	Multiply Carry Miscompare byte 5	PFSALN(49)	PFS89/3
13	Multiply Carry Miscompare byte 6	PFSALN(50)	PFS89/4
14	Multiply Carry Miscompare byte 7	PFSALN(51)	PFS89/5
15	Multiply Carry Miscompare byte 8	PFSALN(52)	PFS89/6
16	Multiply C Register Parity Error byte 0	PFSALN(0)	PFS88/16
17	Multiply C Register Parity Error byte 1	PFSALN(1)	PFS88/17
18	Multiply C Register Parity Error byte 2	PFSALN(2)	PFS88/18
19	Multiply C Register Parity Error byte 3	PFSALN(3)	PFS88/19

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
20	Multiply C Register Parity Error byte 4	PFSALN(4)	PFS88/20
21	Multiply C Register Parity Error byte 5	PFSALN(5)	PFS88/21
22	Multiply C Register Parity Error byte 6	PFSALN(6)	PFS88/22
23	Multiply C Register Parity Error byte 7	PFSALN(7)	PFS88/23
24	Multiply B Register Parity Error byte 0	PFSALN(8)	PFS88/24
25	Multiply B Register Parity Error byte 1	PFSALN(9)	PFS88/25
26	Multiply B Register Parity Error byte 2	PFSALN(10)	PFS88/26
27	Multiply B Register Parity Error byte 3	PFSALN(11)	PFS88/27
28	Multiply B Register Parity Error byte 4	PFSALN(12)	PFS88/28
29	Multiply B Register Parity Error byte 5	PFSALN(13)	PFS88/29
30	Multiply B Register Parity Error byte 6	PFSALN(14)	PFS88/30
31	Multiply B Register Parity Error byte 7	PFSALN(15)	PFS88/31
32	DAI Data Parity Error byte 0	PFSOPI(27)	PFS84/48
33	DAI Data Parity Error byte 1	PFSOPI(28)	PFS84/49
34	DAI Data Parity Error byte 2	PFSOPI(29)	PFS84/50
35	DAI Data Parity Error byte 3	PFSOPI(30)	PFS84/51
36	DAI Data Parity Error byte 4	PFSOPI(31)	PFS84/52
37	DAI Data Parity Error byte 5	PFSOPI(32)	PFS84/53
38	DAI Data Parity Error byte 6	PFSOPI(33)	PFS84/54
39	DAI Data Parity Error byte 7	PFSOPI(34)	PFS84/55
40	ALN Micrand Parity Error byte 0	PFSALN(54)	PFS89/8
41	ALN Micrand Parity Error byte 1	PFSALN(55)	PFS89/9
42	ALN Micrand Parity Error byte 2	PFSALN(56)	PFS89/10
43	ALN Micrand Parity Error byte 3	PFSALN(57)	PFS89/11
44	ALN Micrand Parity Error byte 4	PFSALN(58)	PFS89/12
45	ALN Micrand Parity Error byte 5	PFSALN(59)	PFS89/13
46	ALN Micrand Parity Error byte 6	PFSALN(60)	PFS89/14
47	ALN Micrand Parity Error byte 7	PFSALN(61)	PFS89/15
48	Not Used		
49	Not Used		

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
50	Not Used		
51	Not Used		
52	Processor Did Not Halt Flag from PP Program		
53	Control Store Address bit 0		CSA/53
54	Control Store Address bit 1		CSA/54
55	Control Store Address bit 2		CSA/55
56	Control Store Address bit 3		CSA/56
57	Control Store Address bit 4		CSA/57
58	Control Store Address bit 5		CSA/58
59	Control Store Address bit 6		CSA/59
60	Control Store Address bit 7		CSA/60
61	Control Store Address bit 8		CSA/61
62	Control Store Address bit 9		CSA/62
63	Control Store Address bit 10		CSA/63

Section 5, Subsection 0

This subsection tests C register entry of copies of C register bits from ADATC.

Hardware Tested:

- CREGA bits 61, 63, 65, 67, 69, and 71 entry from ADATC.
- Zero entry into C register bits 72 and 73 under control of NOT U8.

Method:

Testing of the C register for entry of fanout copies of ADATC bits and entry of zeros uses ALN soft control entry points 0, 1, and 2. Entry point 0 will load the C register from ADATC, do one iteration of multiply, clock the Multiply Partial Summing network, and select MSUM bits 8 through 71 to the Output mux. The multiply unit is then halted. Entry point 1 is used to output MSUM bits 8 through 71. Entry point 2 selects MSUM bits 0 through 7 to the Output mux and is used to output these bits.

The microand sequence used will load 64-bit signed integers into the B and C registers. The B register will be loaded with a constant value and the test pattern will be sent to the C register to check the C register entry.

Microands Used:

Sequence Number	Description
10	Run multiply from ALN soft control entry points 0, 1, and 2.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0C; multiplier.
a. Report test pattern; informational.

C.2 Test C register entry of CREGA bits.
a. Start processor at Microand Sequence 10; run multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 in bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0001	00	0000 F578 F578 0000
1	0000 0000 0000 0004	00	0003 D5E3 D5E0 0000
2	0000 0000 0000 0010	00	000F 578F 5780 0000
3	0000 0000 0000 0040	00	003D 5E3D 5E00 0000
4	0000 0000 0000 0100	00	00F5 78F5 7800 0000
5	0000 0000 0000 0400	FF	FC2A 1C2A 2000 0000
6	8000 0000 0000 0010	FF	FFF0 A870 A880 0000

Section 5, Subsection 1

This subsection tests C register entry of sign into bits 72 and 73 under 170 mode control (U8).

Hardware Tested:

- C register entry to bits 72 and 73.
- C register sign select of ADATC bit 3 under control of U8.

Method:

The testing of the sign entry into C register bits 72 and 73 under 170 mode control will done using ALN soft control entry points 0, 1, and 2. Entry point 0 will enter the C register with a sign extended multiplier, do one iteration of multiply, clock the Multiply Partial Summing network, and select MSUM bits 8 through 71 to the Output mux. The multiply unit halts in this state. Entry point 1 is used to output MSUM bits 8 through 71 and entry point 2 selects MSUM bits 0 through 7 to the Output mux and is used to output these bits.

The micrand sequence used will load the C register with 48 bits sign extended upper and lower under U8 control. The B register is loaded with a constant value and the test pattern will be sent to the C register to check the C register entry.

Micrands Used:

Sequence Number	Description
19	Run 170 multiply from ALN soft control entry points 0, 1, and 2.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0C; multiplier.
a. Report test pattern; informational.

C.2 Test C register 170 sign entry.
a. Start processor at Micrand Sequence 19; run 170 multiply from ALN soft control entry points 0, 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 in bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0001	00	0000 F578 F578 0000
1	0000 0000 0000 0002	00	0001 EAF1 EAF0 0000
2	0000 0000 0000 0004	00	0003 D5E3 D5E0 0000
3	0000 0000 0000 0010	00	000F 578F 5780 0000
4	0000 0000 0000 0040	00	003D 5E3D 5E00 0000
5	0000 0000 0000 0100	00	00F5 78F5 7800 0000
6	0000 0000 0000 0400	FF	FC2A 1C2A 2000 0000
7	1000 0000 0000 0010	00	0010 4D08 4CF8 0000
8	2000 0000 0000 0010	00	000F 578F 5780 0000
9	4000 0000 0000 0010	FF	FFF0 A870 A880 0000
A	8000 0000 0000 0010	00	000F 578F 5780 0000
B	5000 0000 0000 0100	FF	FF09 9191 9288 0000

Section 5, Subsection 2

This subsection tests B register sign selection and entry under 170 mode control (U8).

Hardware Tested:

- B register sign selection logic.
- B register entry control and sign entry into bit 64.
- U8 control of multiply decoders complement correction signals for correction on 0X decode and not on 1x decode.

Method:

Testing of the B register sign selection logic in 170 mode, its entry into the B register, and U8 control of the multiply decoders is done in the following manner.

Under soft control provided by soft control entry point 0, the B Operand is multiplied by the lower 10 bits of the multiplier and clocked through the multiply Partial Summing network. The Multiply/Divide Output mux enables MSUM bits 8 through 71 to its output. Soft control entry point 1 is used when outputting MSUM bits 8 through 71 through the ALN General network. Soft control entry point 2 selects MSUM bits 0 through 7 through the Multiply/Divide Output mux and is used to output these bits through the ALN General network.

The major cycle control will select ADATB bit 1 as the sign bit, extend it into bits 0 through 15 and bit 64 of the B register and cause the equation for Complement correction to equal "COMP is not equal to 2s CORR" when "B is negative, and a decode of positive or negative 1x or 2x." The multiplier pulls a -2x from C register bits 71 through 73 and a 1x in from C register bits 69 through 71.

Microrands Used:

Sequence Number	Description
20	Run 170 multiply B register entry from ALN soft control entry points 0, 1, and 2.

Initialization:

- Write multiplier, 0000 0000 0000 0001(HEX), in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern (multiplicand) in RF0B.
  a. Report test pattern; informational.

C.2 Test B register under 170 mode control.
  a. Start processor at Microrand Sequence 20; run 170 multiply B register entry from ALN soft control entry points 0, 1, and 2.
  b. Wait halt.
  c. Read and report contents of RF0A (MSUM bits 8 through 71).
  d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
  a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
  a. Error if not equal to 0000 0000 0000 0003(HEX).

Conditions:

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Condition 2 reports bits 8 through 71 in bits 0 through 63. Condition 3 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	8000 0000 0000 0000	00	0000 0000 0000 0000
1	BFFF 0000 0000 0000	00	0000 0000 0000 0000
2	0000 0000 8000 0000	00	0000 0001 0000 0000
3	4000 0000 8000 0000	00	0001 FFFE FFFF FFFE
4	4000 7FFF FFFF FFFF	00	0001 0000 0000 0000
5	4000 0000 0000 0000	00	0001 FFFF FFFF FFFE
6	4000 0000 0000 0001	00	0001 FFFF FFFF FFFC

Section 5, Subsection 3

This subsection tests Complement Correction Signals for proper operation under 170 mode control (U8).

Hardware Tested:

- Multiplier decoders for complement correction control from U8.

Method:

The testing of the multiply decoders for complement correction operation in 170 mode is done using the following controls.

ALN soft control entry point 10(HEX) will load and shift the C register but the multiply Partial Summing network will not be clocked. This effectively discards the multiply of the lower 10 bits of the multiplier and the 170 sign extension to the lower 2 bits of the C register. The next 12 bits of the C register are then used as the multiplier and the result is clocked into the multiply Partial Summing network and enabled through the Output mux as MSUM bits 8 through 71.

Soft control entry point 1 is used when outputting MSUM bits 8 through 71 through the ALN General network. Soft control entry point 2 selects MSUM bits 0 through 7 through the Output mux and is used when outputting these bits through the ALN General network.

The major cycle control will load the B and C registers with 170 sign extension and cause the equation for CORR in the multiply decoders to be changed to the following:

- CORR is not equal to COMP when B is negative and the decode is equal to 1x or 2x.

This subsection runs two passes through the test patterns: first with a positive multiplicand and then with a negative multiplicand. The test patterns are used as the multiplier.

Microands Used:

Sequence Number	Description
21	Run 170 multiply for CORR check from ALN soft control entry points 10(HEX), 1, and 2.

Initialization:

- Write positive multiplicand , 0000 7ABC 7ABC 0000(HEX), in RF0B if iteration count is less that 14(HEX).
- Write negative multiplicand , 4000 7ABC 7ABC 0000(HEX), in RF0B if iteration count is greater than 13(HEX).

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0C.
a. Report test pattern; informational.

C.2 Test CORR signals.
a. Start processor at Microrand Sequence 21; run 170 multiply for 2s CORR check from ALN soft control entry points 10(HEX), 1, and 2.
b. Wait halt.
c. Read and report contents of RF0A (MSUM bits 8 through 71).
d. Error if not equal to expected result.

C.3 Read contents of RF09 bits 48 through 55 (MSUM bits 0 through 7) and report them in byte 7.
a. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Expected result for each pattern is shown for both the positive and negative multiplicand. Condition 3 reports bits 8 through 71 in bits 0 through 63. Condition 4 reports bits 0 through 7 in byte 7.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
0	0000 0000 0000 0400	00	0000 7ABC 7ABC 0000
1	0000 0000 0000 1000	00	0000 F578 F578 0000
2	0000 0000 0000 4000	00	0003 D5E3 D5E0 0000
3	0000 0000 0001 0000	00	000F 578F 5780 0000
4	0000 0000 0004 0000	00	003D 5E3D 5E00 0000
5	0000 0000 0010 0000	00	00F5 78F5 7800 0000
6	0000 0000 0040 0000	FF	FC2A 1C2A 2000 0000
7	F800 0000 0000 0400	FF	FFFF 8543 8544 0000
8	F800 0000 0000 1000	FF	FFFF 0A87 0A88 0000
9	F800 0000 0000 4000	FF	FFFC 2A1C 2A20 0000
A	F800 0000 0001 0000	FF	FFF0 A870 A880 0000
B	F800 0000 0004 0000	FF	FFC2 A1C2 A200 0000
C	F800 0000 0010 0000	FF	FF0A 870A 8800 0000
D	F800 0000 0040 0000	00	03D5 E3D5 E000 0000
E	0000 0000 0000 1C00	00	0001 EAF1 EAF0 0000
F	0000 0000 0000 7000	00	0006 B64E B648 0000
10	0000 0000 0001 C000	00	001A D93A D920 0000
11	0000 0000 0007 0000	00	006B 64EB 6480 0000
12	0000 0000 001C 0000	00	01AD 93AD 9200 0000
13	0000 0000 0070 0000	FF	FF0A 870A 8800 0000
14	0000 0000 0000 0400	00	0000 8543 8543 FFFF
15	0000 0000 0000 1000	00	0001 0A87 0A87 FFFE
16	0000 0000 0000 4000	00	0004 2A1C 2A1F FFF8
17	0000 0000 0001 0000	00	0010 A870 A87F FFE0
18	0000 0000 0004 0000	00	0042 A1C2 A1FF FF80
19	0000 0000 0010 0000	00	010A 870A 87FF FE00

Iteration Count	Test Pattern	Expected Result for Bits 0 through 7	Expected Result for Bits 8 through 71
1A	0000 0000 0040 0000	FF	FBD5 E3D5 E000 0800
1B	F800 0000 0000 0400	FF	FFFF 7ABC 7ABC 0001
1C	F800 0000 0000 1000	FF	FFFE F578 F578 0002
1D	F800 0000 0000 4000	FF	FFFB D5E3 D5E0 0008
1E	F800 0000 0001 0000	FF	FFEF 578F 5780 0020
1F	F800 0000 0004 0000	FF	FFBD 5E3D 5E00 0080
20	F800 0000 0010 0000	FF	FEF5 78F5 7800 0200
21	F800 0000 0040 0000	00	042A 1C2A 1FFF F800
22	0000 0000 0000 1C00	00	0002 150E 150F FFFC
23	0000 0000 0000 7000	00	0007 49B1 49B7 FFF2
24	0000 0000 0001 C000	00	001D 26C5 26DF FFC8
25	0000 0000 0007 0000	00	0074 9B14 9B7F FF20
26	0000 0000 001C 0000	00	01D2 6C52 6DFF FC80
27	0000 0000 0070 0000	FF	FEF5 78F5 7800 0200

SECTION 6 - CHECKS OF C REGISTER ENTRY, MULTIPLY/DIVIDE CONTROL AND SIGN SIGNALS TO THE ALN GENERAL NETWORK AND THE MAJOR CYCLE CONTROL BUFFER

This section verifies the operation of the following areas of the logic: C register entry of MSUM bits, Integer multiply overflow detection and reporting, Multiply/Divide operation control in the ALN General network, Multiply/Divide sign selection logic and its usage in the ALN General network, and the Multiply/Divide Major Cycle Control buffer.

Subsection 0 - Test the entry and shifting of MSUM bits 60 through 71 through the C register.

Subsection 1 - Test integer multiply overflow and the C register and multiply Final Adder zero checks.

Subsection 2 - Test 32-bit integer multiply overflow detection.

Subsection 3 - Test Multiply/Divide unit for generation of integer Multiply/Divide overflow and the ALN General network for encoding the overflow in the Normalize encoder.

Subsection 4 - Test B and C register sign selection logic to form Multiply/Divide sign and the enable of the Multiply/Divide sign to the B and C sign fanouts that are sent to the ALN General network.

Subsection 5 - Test sequencing of the Multiply/Divide sign to the ALN General network sign in the Microand register and the enable of the ALN General network sign to the C sign fanouts.

Subsection 6 - Test C register bit 7 XORed with Multiply/Divide sign and extended through the ALN General network shifter.

Subsection 7 - Test Major Cycle Control buffer for holding of U4, U6, and U8.

Subsection 8 - Test Major Cycle Control buffer for holding of U1, U2, and U3.

Method:

Refer to Subsection Method for details.

All conditions in this subsection that report status will read/format and report all status listed in the Section Status Format.

Section Status Format:

The following bit assignments are used by all conditions that report status.

NOTE

Any status not read by a condition and all unused bit positions will be reported as zero. Refer to Section Method for status that will be read and reported.

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
0	Soft Control Register Parity Error byte 0	PFSALN(62)	PFS88/14
1	Soft Control Register Parity Error byte 1	PFSALN(63)	PFS88/15
2	Not Used		
3	Not Used		
4	Not Used		
5	Not Used		
6	Multiply Carry Miscompare byte 0	PFSALN(44)	PFS88/60
7	Multiply Carry Miscompare byte 1	PFSALN(45)	PFS88/61
8	Not Used		
9	Multiply Carry Miscompare byte 2	PFSALN(46)	PFS89/0
10	Multiply Carry Miscompare byte 3	PFSALN(47)	PFS89/1
11	Multiply Carry Miscompare byte 4	PFSALN(48)	PFS89/2
12	Multiply Carry Miscompare byte 5	PFSALN(49)	PFS89/3
13	Multiply Carry Miscompare byte 6	PFSALN(50)	PFS89/4
14	Multiply Carry Miscompare byte 7	PFSALN(51)	PFS89/5
15	Multiply Carry Miscompare byte 8	PFSALN(52)	PFS89/6
16	Multiply C Register Parity Error byte 0	PFSALN(0)	PFS88/16
17	Multiply C Register Parity Error byte 1	PFSALN(1)	PFS88/17
18	Multiply C Register Parity Error byte 2	PFSALN(2)	PFS88/18
19	Multiply C Register Parity Error byte 3	PFSALN(3)	PFS88/19

Report Bit Number	Status Reported	Signal Name	Register/Bit
20	Multiply C Register Parity Error byte 4	PFSALN(4)	PFS88/20
21	Multiply C Register Parity Error byte 5	PFSALN(5)	PFS88/21
22	Multiply C Register Parity Error byte 6	PFSALN(6)	PFS88/22
23	Multiply C Register Parity Error byte 7	PFSALN(7)	PFS88/23
24	Multiply B Register Parity Error byte 0	PFSALN(8)	PFS88/24
25	Multiply B Register Parity Error byte 1	PFSALN(9)	PFS88/25
26	Multiply B Register Parity Error byte 2	PFSALN(10)	PFS88/26
27	Multiply B Register Parity Error byte 3	PFSALN(11)	PFS88/27
28	Multiply B Register Parity Error byte 4	PFSALN(12)	PFS88/28
29	Multiply B Register Parity Error byte 5	PFSALN(13)	PFS88/29
30	Multiply B Register Parity Error byte 6	PFSALN(14)	PFS88/30
31	Multiply B Register Parity Error byte 7	PFSALN(15)	PFS88/31
32	DAI Data Parity Error byte 0	PFSOPI(27)	PFS84/48
33	DAI Data Parity Error byte 1	PFSOPI(28)	PFS84/49
34	DAI Data Parity Error byte 2	PFSOPI(29)	PFS84/50
35	DAI Data Parity Error byte 3	PFSOPI(30)	PFS84/51
36	DAI Data Parity Error byte 4	PFSOPI(31)	PFS84/52
37	DAI Data Parity Error byte 5	PFSOPI(32)	PFS84/53
38	DAI Data Parity Error byte 6	PFSOPI(33)	PFS84/54
39	DAI Data Parity Error byte 7	PFSOPI(34)	PFS84/55
40	ALN Micrand Parity Error byte 0	PFSALN(54)	PFS89/8
41	ALN Micrand Parity Error byte 1	PFSALN(55)	PFS89/9
42	ALN Micrand Parity Error byte 2	PFSALN(56)	PFS89/10
43	ALN Micrand Parity Error byte 3	PFSALN(57)	PFS89/11
44	ALN Micrand Parity Error byte 4	PFSALN(58)	PFS89/12
45	ALN Micrand Parity Error byte 5	PFSALN(59)	PFS89/13
46	ALN Micrand Parity Error byte 6	PFSALN(60)	PFS89/14
47	ALN Micrand Parity Error byte 7	PFSALN(61)	PFS89/15
48	Not Used		
49	Not Used		

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
50	Not Used		
51	Not Used		
52	Processor Did Not Halt Flag from PP Program		
53	Control Store Address bit 0		CSA/53
54	Control Store Address bit 1		CSA/54
55	Control Store Address bit 2		CSA/55
56	Control Store Address bit 3		CSA/56
57	Control Store Address bit 4		CSA/57
58	Control Store Address bit 5		CSA/58
59	Control Store Address bit 6		CSA/59
60	Control Store Address bit 7		CSA/60
61	Control Store Address bit 8		CSA/61
62	Control Store Address bit 9		CSA/62
63	Control Store Address bit 10		CSA/63

Section 6, Subsection 0

This subsection tests the entry and shifting of MSUM bits 60 through 71 through the C register.

Hardware Tested:

- C register bits 0 through 11 entry.

Method:

The testing of the C register entry of MSUM bits 60 through 71 is done in the following manner.

The B and C registers are entered with 64 bits. The result of the first iteration is discarded by not enabling the clock on the Multiply Partial Summing network. Then the C register is shifted and the second iteration is saved by clocking the Multiply Partial Summing network. Two minor cycle no-ops are performed to allow the results to show up on the output of the Multiply Final Adder. Then the C register is shifted and MSUM bits 60 through 71 are entered into C register bits 0 through 11. The C register is again shifted to move these bits into C register bits 12 through 23 and MSUM bits 60 through 71 are again entered into C register bits 0 through 11. C register bits 8 through 71 are then output through the ALN General network and bits 8 through 23 will be checked for the expected value.

All status listed in the Section Status Format will be read and reported by this subsection.

Initialization:

- Write the following ALN soft control memory map (hexadecimal) into the locations specified.

Location	32-bit Value to be Written
0	4108 0000
1	D308 0000
2	0108 0000
3	0108 0000
4	C108 0000
5	C108 0000
6	0108 0000
7	0000 0000

- Write multiplier for this subsection, 0000 0000 0000 0800(HEX), in RF0C.

Microrands Used:

Sequence Number	Description
22	Run multiply and Output the C register.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test shifting of MSUM through C register.
a. Start processor at Microrand Sequence 22; run multiply and output through the C register.
b. Wait halt.
c. Read and report contents of RF0A bits 0 through 15 (C register bits 8 through 23).
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Only bits 0 through 15 from the register file are reported. Bits 16 through 63 are masked to zero by the PP program.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 15
0	0000 0000 0000 0000	0000
1	0000 0000 0000 0FFF	FFFF
2	0000 0000 0000 0AAA	AAAA
3	0000 0000 0000 0555	5555
4	0000 0000 0000 0123	3123
5	0000 0000 0000 0DCD	DDCD

Section 6, Subsection 1

This subsection tests integer multiply overflow and performs the C register and Multiply Final Adder zero checks.

Hardware Tested:

- ALN Exponent Aux Board for enabling and reporting of integer multiply overflow.
- Multiply/Divide unit for detection of C register bits 4, 5, and 6 for zero and nonzero. C register bit 7 equal to zero when MD sign is zero.
- Multiply Final Adder arrays zero test logic.
- Multiply/Divide sign logic.
- Multiply/Divide run signal select of Major Cycle Control buffer to Major Cycle Control register.
- Major Cycle Control buffer for UBUF bits 5 and 12.

Method:

The testing of the logic that checks for and reports integer overflow to the User Condition register is performed in the following manner.

The ALN soft control will cause the following sequence of operations in the multiply unit. The C register will be enabled and loaded, and the Multiply Partial Summing network clock enabled. Then the C register is shifted and the Multiply Partial Summing network clock is enabled, but the feedback path is not enabled. Two minor cycle no-ops are run to allow time for the results of the first multiply to be seen on the output of the Final Adder.

Then the Final Adder bits 60 through 71 will be shifted into the C register bits 0 through 11 and the results of the second multiply will show up on MSUM bits 0 through 71, which will be enabled to the Output mux.

The major cycle control will enable loading the B and C registers and cause the ALN soft control to execute. The third micrand of the sequence will copy the Multiply/Divide output mux through the ALN General network and at the same time enable the ALN Exponent Aux to sense Integer Multiply overflow and report it to the User Condition register bit 57.

The User Condition register will be clocked and the micrand sequence will copy the UCR to RF0D.

Only bits 55 and 57 through 62 of the User Condition register in RF0D will be checked for the proper result.

Isolation Notes:

The test patterns will check the following areas of the logic.

- Test pattern 0 tests that integer multiply overflow can be reported.
- Test patterns 1 through 4 test for C register equal to nonzero.
- Test patterns 5 through 48(HEX) test the Final Adder zero test logic for nonzero detection.
- Test pattern 49(HEX) checks that all logic reports zeros and is not isolatable beyond the fact that one of areas of the logic did not report zeros or the AND Gates are not working properly.
- Test pattern 4A(HEX) tests U5 overriding of MSUM bits 63 through 71 nonzero.
- Test pattern 4B(HEX) tests MDSNG blocking nonzero check of C register bit 7.

All status in the Section Status Format will be reported.

Microrands Used:

Sequence Number	Description
23	Clear UCR.
24	Run 64-bit integer multiply overflow and read/clear UCR.

Initialization:

- Write the following ALN soft control memory map (hexadecimal) into the locations specified.

Location	32-bit Value to be Written
0	5108 0100
1	D308 0000
2	0108 0000
3	C108 0000
4	0108 0000
5	0114 0000
6	0000 0000
7	0000 0000

- Write 0000 0000 0000 0000(HEX) in RF09.
- Write multiplier for Test Pattern 0, 0000 0000 0000 0808(HEX), in RF0C.
- Start processor at Microrand Sequence 23 (clear UCR).

Conditions:

The following conditions are repeated for each test pattern.

If Test Pattern 1, write new multiplier, 0000 0000 0000 0008(HEX), in RF0C.

If Test Pattern 5, write new multiplier, 0000 0000 0008 0000(HEX), in RF0C.

If Test Pattern 12(HEX), write new multiplier, 0000 0000 0000 0800(HEX), in RF0C.

If Test Pattern 4B, write new multiplier, 0000 0000 0000 0080(HEX), in RF0C.

C.0 Report iteration count; informational.

C.1 Report multiplier; informational.

Conditions:

C.2 Write test pattern in RF0B.
a. Report test pattern.

C.3 Test 64-bit integer multiply overflow.
a. Start processor at Microrand Sequence 24; run 64-bit integer multiply overflow and read/clear UCR.
b. Wait halt.
c. Read and report contents of RF0D bits 55, 57 through 62. (On test patterns 0 through 48(HEX), report error if not equal to 0000 0000 0000 0040(HEX). On test patterns 49(HEX) through 4B(HEX), report error if not equal to all zeros).

C.4 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.3.

Exit subsection if no more test patterns.

Test Patterns (Hexadecimal):

Iteration Count	Test Pattern
0	7FFF FFFF FFFF FFFF
1	0000 0000 0000 0004
2	0000 0000 0000 0008
3	0000 0000 0000 0010
4	0000 0000 0000 0020
5	7FFF FFFF FFFF FFFF
6	4000 0000 0000 0000
7	2000 0000 0000 0000
8	1000 0000 0000 0000
9	0800 0000 0000 0000
A	0400 0000 0000 0000
B	0200 0000 0000 0000

Iteration Count	Test Pattern
C	0100 0000 0000 0000
D	0080 0000 0000 0000
E	0040 0000 0000 0000
F	0020 0000 0000 0000
10	0010 0000 0000 0000
11	0008 0000 0000 0000
Start of Sequence for Test Pattern:	
12	0000 0000 0000 0100
13	0000 0000 0000 0200
14	0000 0000 0000 0400
15	0800 0000 0000 0800
.	.
.	.
.	.
45	0800 0000 0000 0000
46	1000 0000 0000 0000
47	2000 0000 0000 0000
48	4000 0000 0000 0000
End of Sequence for Test Pattern:	
49	0000 0000 0000 0000
4A	0000 0000 0000 00FF
4B	FFFF FFFF FFFF FFFC

Section 6, Subsection 2

This subsection tests 32-bit integer multiply overflow Detection.

Hardware Tested:

- U5 control of sampled C register bits and Final Adder bits for overflow detection.
- Final Adder bits 64 through 71 for zero detection.
- Not U12 disable of sampled C register bits for overflow detection.

Method:

The testing of major cycle control of which bits of the C register and Final Adder are checked for overflow detection is performed in the following manner.

The ALN soft control will cause three separate multiplies to be performed. Each of the three multiplies will be separate as the feedback in the Multiply Partial Summing network will not be enabled. After the three multiplies have been started, one minor cycle no-op is performed to allow time for the first result to appear in the Final Adder. These results are then shifted through the C register to the proper bit positions for 32- or 64-bit overflow detection.

The major cycle controls used by this subsection will cause either a 64-bit multiply (U12) overflow detection or a 32-bit multiply (Not U5) overflow detection. Integer overflow reporting will be enabled in the ALN Exponent Aux board, the User Condition register will be clocked and will then be copied to RF0D.

The User Condition register bits 55, 57 through 62 as copied into RF0D will be reported and checked for the proper value.

The three different multipliers are used to position the result in the proper bit positions to be checked.

Micrands Used:

Sequence Number	Description
23	Clear UCR.
24	Run 64-bit integer multiply overflow and read/clear UCR.
25	Run 32-bit integer multiply overflow and read/clear UCR.

Initialization:

- Write the following ALN soft control memory map (hexadecimal) into the locations specified.

Location	32-bit Value to be Written
0	5108 0100
1	D308 0000
2	D108 0000
3	C108 0000
4	C108 0000
5	0114 0000
6	0000 0000
7	0000 0000
8	C108 0000
9	C108 0000
A	C108 0000
B	0108 0000

- Write 0000 0000 0000 0000(HEX) in RF09.
- Write the multiplier for patterns 0 through 9, 0000 0000 0000 0800 (HEX), in RF0C.
- Start processor at Microrand Sequence 23 (clear UCR).

Conditions:

The following conditions are repeated for each test pattern.

If Test Pattern A(HEX), write new multiplier, 0000 0000 0080 0000(HEX), in RF0C.

C.0 Report iteration count; informational.

C.1 Report multiplier; informational.

C.2 Write test pattern in RF0B.
a. Report test pattern; informational.

C.3 Check 32-bit integer multiply overflow.
a. Start processor at Microrand Sequence 25; run 32-bit integer multiply overflow and read/clear UCR.
b. Wait halt.
c. Read contents of RF0D bits 55, 57 through 62. Mask all other bits to zero.
d. Report result. Error if not equal to expected result.

C.4 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

Conditions:

C.5 Check 64-bit integer multiply overflow.
a. Start processor at Microrand Sequence 24; run 64-bit integer multiply overflow and read/clear UCR.
b. Wait halt.
c. Read contents of RF0D bits 55, 57 through 62. Mask all other bits to zero.
d. Report result. Error if not equal to expected result.

C.6 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.7 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.3.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Only expected result bits 56 through 63 are shown. All others are expected as zero.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 56 through 63
0	0000 0000 0000 0800	C.3	40
		C.5	00
1	0000 0000 0000 0400	C.3	40
		C.5	00
2	0000 0000 0000 0200	C.3	40
		C.5	00
3	0000 0000 0000 0100	C.3	40
		C.5	00
4	7FFF FFFF FFFF F00E	C.3	00
		C.5	00
5	0000 0000 0000 0010	C.3	00
		C.5	40

Iteration Count	Test Pattern	Conditions	for Bits 56 through 63
6	0000 0000 0000 0020	C.3 C.5	00 40
7	0000 0000 0000 0040	C.3 C.5	00 40
8	0000 0000 0000 0080	C.3 C.5	00 40
9	0000 0000 0000 0000	C.3 C.5	00 00
A	0000 0000 0000 0001	C.3 C.5	40 00
B	0000 0000 0000 0002	C.3 C.5	40 00
C	0000 0000 0000 0004	C.3 C.5	40 00
D	0000 0000 0000 0008	C.3 C.5	40 00
E	0000 0000 0000 0010	C.3 C.5	40 00
F	0000 0000 0000 0020	C.3 C.5	40 00
10	0000 0000 0000 0040	C.3 C.5	40 00
11	0000 0000 0000 0080	C.3 C.5	40 00
12	0000 0000 0000 0100	C.3 C.5	40 40

Section 6, Subsection 3

This subsection tests Multiply/Divide unit for generation of Integer Multiply Divide overflow and ALN General network for encoding of the overflow in the Normalize encoder.

Hardware Tested:

- Multiply/Divide unit for MFOVF Signal from the following:
  - MPY80 and MSUM bit 23.
  - MPY70 and MSUM bit 23 and B and C normalized.
  - MPY70 and not B and C normalize.
- Multiply/Divide unit for disable of MPY70 and MPY80 when Minor Cycle Control V10 is active.
- ALN General network Normalize encoder for encoding of MFOVF and generating the proper shift count to the shifter.

Method:

The testing of the MFOVF generation and encoding is performed in the following manner.

Both ALN soft control entry points used will generate MSUM bit 23 if bit 15 is set in the multiplicand and the C Operand bits 32 through 63 are equal to 00000800(HEX). Both will also enable MSUM bits 8 through 71 to the Multiply/Divide Output mux. Entry point 0 has U10 clear and entry point 10(HEX) has U10 set to disable MPY70 and MPY80.

All sequences used in this subsection will use the same controls to input to the Multiply unit and encode and output the multiply result except for the C170 Format flag and the ALN soft control entry point used in the first micrand.

The first micrand loads the B and C registers of the multiply unit and the multiply starts.

The second micrand causes the following in the ALN General network while holding the output of the Multiply/Divide unit Static.

- The Shift mux is loaded with MSUM bits 8 through 71 into bits 0 through 63 and the operand from RF0C into bits 64 through 127 (Mux Code = 7).
- The Normalize encoder encodes the MFOVF Signal generating a Shift Count of 49 if it is active or of 48 if it is inactive.
- The shifter shifts according to the Normalize encoder.
- The output is taken from the Shift network bits 64 through 127 and written in the Register File.

All status listed in the Section Status will be read and reported.

Microrands Used:

Sequence Number	Description
26	Run 180 multiply and encode MFOVF from ALN soft control entry points 0 and 1.
27	Run 170 multiply and encode MFOVF from ALN soft control entry points 0 and 1.
28	Run 180 multiply and encode MFOVF from ALN soft control entry points 10(HEX) and 1, disable MPY80.
29	Run 170 multiply and encode MFOVF from ALN soft control entry points 10(HEX) and 1, disable MPY70.

Initialization:

- Write the following ALN soft control memory map (hexadecimal) in the locations specified.

Location	32-bit Value to be Written
0	4108 4108
1	D308 D328
2	0108 0128
3	0000 0108
4	0114 0114
5	0000 0000
6	0000 0000
7	0000 0000

Conditions:

The following conditions are repeated for each test.

C.0 Report iteration count; informational.

C.1 Write multiplier in RF0C.
a. Report multiplier; informational.

C.2 Write multiplicand in RF0B.
a. Report multiplicand; informational.

C.3 Test 180 mode overflow.
a. Start processor at Microrand Sequence 26; run 180 multiply and Encode MFOVF from ALN soft control entry points 0 and 1.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

Conditions:

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test 170 mode overflow.
a. Start processor at Microrand Sequence 27; run 170 multiply and encode MFOVF from ALN soft control entry points 0 and 1.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

C.6 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.7 Test disable MPY80.
Start processor at Microrand Sequence 28; run 180 multiply and encode MFOVF from ALN soft control entry points 10(HEX) and 1, disable MPY80.
a. Wait halt.
b. Read and report contents of RF0A.
c. Error if not equal to expected result.

C.8 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.9 Test disable MPY70.
Start processor at Microrand Sequence 29; run 170 multiply and encode MFOVF from ALN soft control entry points 10(HEX) and 1, disable MPY70.
a. Wait halt.
b. Read and report contents of RF0A.
c. Error if not equal to expected result.

C.10 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.3.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

Iteration Count	Multiplier	Multiplicand	Condition	Expected Result
0	0248 0000 0000 0800	0001 0000 0000 0000	C.3	8000 0000 0000 0124
			C.5	8000 0000 0000 0124
			C.7	0000 0000 0000 0248
			C.9	0000 0000 0000 0248
1	0248 8000 0000 0800	0001 8000 0000 0000	C.3	C000 0000 0000 0124
			C.5	C000 0000 0000 0124
			C.7	8000 0000 0000 0248
			C.9	8000 0000 0000 0248
2	0248 8000 0000 0800	0000 8000 0000 0000	C.3	8000 0000 0000 0248
			C.5	8000 0000 0000 0248
			C.7	8000 0000 0000 0248
			C.9	8000 0000 0000 0248
3	2248 8000 0000 0800	0000 8000 0000 0000	C.3	8000 0000 0000 2248
			C.5	4000 0000 0000 1124
			C.7	8000 0000 0000 2248
			C.9	8000 0000 0000 2248
4	2248 0000 0000 0800	0000 8000 0000 0000	C.3	8000 0000 0000 2248
			C.5	8000 0000 0000 2248
			C.7	8000 0000 0000 2248
			C.9	8000 0000 0000 2248
5	0248 8000 0000 0800	2000 8000 0000 0000	C.3	8000 0000 0000 0248
			C.5	4000 0000 0000 0124
			C.7	8000 0000 0000 0248
			C.9	8000 0000 0000 0248
6	0248 8000 0000 0800	2000 0000 0000 0000	C.3	0000 0000 0000 0248
			C.5	0000 0000 0000 0248
			C.7	0000 0000 0000 0248
			C.9	0000 0000 0000 0248
7	2248 0000 0000 0800	2001 0000 0000 0000	C.3	8000 0000 0000 1124
			C.5	8000 0000 0000 1124
			C.7	0000 0000 0000 2248
			C.9	0000 0000 0000 2248

Section 6, Subsection 4

This subsection tests B and C Operand Sign Selection logic to form Multiply/Divide Sign and the enable of the Multiply/Divide sign to the B and C Sign fanouts that are sent to the ALN General network.

Hardware Tested:

- Multiply/Divide Sign logic from B and C Sign inputs.
- B Sign enables from ADATB bit 0 or bit 32.
- C Sign enables from ADATC bit 0 or bit 32.
- U13 disable of sign selection.
- Major Cycle Control buffer for not bits 0, 2, 5 and 13.
- Not U0 enable of Multiply/Divide sign to fanouts.

Method:

The testing of the Multiply/Divide Sign logic and its enables to the B and C Sign Fanouts is performed under major cycle control. The minor cycle control used will only condition the Multiply/Divide Output mux to pass the ADATB operand on to the ALN General network.

The major cycle control from the first micrand of each sequence will select one of three sign determinations.

1. Sign from input bits ADATC 0 and ADATB 0 (U2 and not U13, U5 and not U13).
2. No sign to Multiply/Divide Sign (U2 and U13, U5 and U13).
3. Sign from input bits ADATC 32 and ADATB 32 (Not U2).

The test operands will be sent with the first micrand. The third micrand is used to condition the ALN General network in the following manner.

- The B sign will be extended into Shift mux bits 64 through 111 and the C sign will be extended into Shift mux bits 112 through 127. (Mux Code 8 and C greater than B from SKG Code 9 and C operand exponent field greater than B operand exponent field).
- The output of the Shifter bits 64 through 127 will be written into the register file.

All Status listed in the Section Status Format will be read and checked.

Micrands Used:

Sequence Number	Description
30	Select Multiply/Divide sign from bit 0.
31	Disable select Multiply/Divide sign from bit 0.
32	Select Multiply/Divide sign from bit 32.

Initialization:

- Write the following ALN soft control memory map (hexadecimal) into the locations specified.

Location	32-bit Value to be Written
0	5108 0100
1	D308 0000
2	0108 0000
3	C108 0000
4	0114 0000
5	0000 0000
6	0000 0000
7	0000 0000

- Write 0000 0000 0000 0000(HEX) in RF09.

Conditions:

The following conditions are repeated for each test.

C.0 Report iteration count; informational.

C.1 Write B pattern in RF0B.
a. Report B Pattern.

C.2 Write C pattern RF0C.
a. Report C Pattern.

C.3 Test determine MDSGND from ADATC bit 0 and ADATB bit 0.
a. Start processor at Micrand Sequence 30; Select Multiply/Divide sign from bit 0.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

C.4 Read/format and report status according to Section Status Format.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test disable MGSGND by U13.
a. Start processor at Micrand Sequence 31; disable Select Multiply/Divide sign from bit 0.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

C.6 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

Conditions:

C.7 Test determine MDSGND from bit 32.
a. Start processor at Micrand Sequence 32; Select Multiply/Divide sign from bit 32.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

C.8 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.9 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.3.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

Iteration Count	B Pattern	C Pattern	Condition	Expected Result
0	0000 0000 0000 0000	0001 0000 0000 0000	C.3	0000 0000 0000 0000
			C.5	0000 0000 0000 0000
			C.7	0000 0000 0000 0000
1	8000 0000 0000 0000	0001 0000 0000 0000	C.3	FFFF FFFF FFFF FFFF
			C.5	0000 0000 0000 0000
			C.7	0000 0000 0000 0000
2	0000 0000 0000 0000	8001 0000 0000 0000	C.3	FFFF FFFF FFFF FFFF
			C.5	0000 0000 0000 0000
			C.7	0000 0000 0000 0000
3	8000 0000 0000 0000	8001 0000 0000 0000	C.3	0000 0000 0000 0000
			C.5	0000 0000 0000 0000
			C.7	0000 0000 0000 0000
4	0000 0000 8000 0000	0001 0000 0000 0000	C.3	0000 0000 0000 0000
			C.5	0000 0000 0000 0000
			C.7	FFFF FFFF FFFF FFFF
5	0000 0000 0000 0000	0001 0000 8000 0000	C.3	0000 0000 0000 0000
			C.5	0000 0000 0000 0000
			C.7	FFFF FFFF FFFF FFFF
6	0000 0000 8000 0000	0001 0000 8000 0000	C.3	0000 0000 0000 0000
			C.5	0000 0000 0000 0000
			C.7	0000 0000 0000 0000

Iteration Count	B Pattern	C Pattern	Condition	Expected Result
7	8000 0000 0000 0000	0001 0000 8000 0000	C.3	FFFF FFFF FFFF FFFF
			C.5	0000 0000 0000 0000
			C.7	FFFF FFFF FFFF FFFF
8	0000 0000 8000 0000	8001 0000 0000 0000	C.3	FFFF FFFF FFFF FFFF
			C.5	0000 0000 0000 0000
			C.7	FFFF FFFF FFFF FFFF

Section 6, Subsection 5

This subsection tests sequencing of Multiply/Divide sign to ALN General network Internal sign in the Micrand register and enable of Internal sign to the C sign fanout.

Hardware Tested:

- MDSGNO, Multiply/Divide sign, to ALN General network.
- ALN General network for enable of MDSGNO and disable of C sign and Adder sign to Internal sign during both Multiply/Divide start and run operations.
- Enable of Internal sign to C sign fanout in Multiply/Divide unit.
- Major Cycle Control buffer for holding of U0.

Method:

The testing of entry of the Multiply/Divide sign into Internal sign and its gating to the C Sign fanout in the following micrand during both Multiply/Divide start and run operations is performed under major cycle control. The minor cycle control used will only condition the Multiply/Divide Output mux to pass the ADATB operand on to the ALN General network.

The major cycle control used and the sequencing of the test pattern along with constants will cause the following.

- The Multiply/Divide sign will be the complement of the test pattern sign.
- The sign that is output and checked will have been enabled either on the Multiply/Divide start signal (Micrand Sequence 33) or on the Multiply Divide Run Signal (Micrand Sequence 34).
- Attempt to enter C sign and Large Adder Overflow sign to the ALN General network sign (Micrand bits 24 and 25).

The last micrand of both sequences will cause the C sign fanout to copy Internal Sign through the lower half of the Shift mux and write it into the register file. (SKG Code F(HEX)).

The first micrand of Micrand Sequence 33 also contains control for the ALN General network to cause it to generate Adder sign equal to 1 in addition to the controls for the Multiply/Divide network (mux Code E(HEX), complement code 2 [U5] Carry Code 6, Adder Code of Add).

All status in the Section Status Format will be read and checked.

Micrands Used:

Sequence Number	Description
33	Enter Multiply/Divide sign to Internal sign on Multiply/Divide Start.
34	Enter Multiply/Divide sign to Internal sign on Multiply/Divide Run.

Initialization:

- Write 0000 0000 0000 0000(HEX) in RF09; used by first micrand of Micrand Sequence 34.
- Write FFFF FFFF FFFF FFFF(HEX) in RF0C; used as C Operand by both micrand sequences.

NOTE

This subsection uses the ALN soft control memory map written by section 6, subsection 4.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
  a. Report test pattern; informational.

C.2 Test Enable Sign on Multiply/Divide Start.
  a. Start processor at Micrand Sequence 33; enter Multiply/Divide Sign to Internal Sign on Multiply/Divide Start.
  b. Wait halt.
  c. Read and report contents of RF0A.
  d. Error if not equal to expected result.

Conditions:

C.3 Read/format and report status according to Section Status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Test Enable Sign on Multiply/Divide Run.
a. Start processor at Microand Sequence 34; enter Multiply/Divide Sign to Internal Sign on Multiply/Divide Run.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

C.5 Read/format and report status according to Section Status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

Iteration Count	Test Pattern	Expected Result
0	8000 0000 0000 0000	0000 0000 0000 0000
1	0000 0000 0000 0000	FFFF FFFF FFFF FFFF

Section 6, Subsection 6

This subsection tests C register bit 7 XORed with Multiply/Divide sign and extended through the ALN General network shifter.

Hardware Tested:

- Multiply/Divide unit for C register bit 7 XOR with MDSGND to generate Integer Multiply Divide Overflow (IMD).
- ALN General network for extending of IMD in the Shifter on Multiply/Divide operation.
- ALN General network for block of extend zeros control on a Multiply/Divide operation.

Method:

The testing of C register bit 7 XOR with MDSGND to form IMD and its entry into the Shifter is performed in the following manner.

The first and second microand minor cycle control will cause the results of the first Multiply iteration to be shifted into C register bits 0 through 11 and the Multiply unit will stop.

The major cycle control will load the B and C registers and hold the Multiply/Divide sign stable for use by the third microand.

The third microand will use the sign value held in RF09 and the multiplier sign (always zero) to generate MDSGND. This will be XORed with C register bit 7 and sent to the ALN General network shifter. The shifter will shift 63 places to extend this value through the shifter bits 0 through 62.

The complement of shifter bits 16 through 63 will be added to zeros in the adder bits 0 through 47 and output to the Register File in bits 16 through 63. Bits 0 through 15 come from the shifter but will not be reported or checked.

All status in the Section Status Format will be read and checked.

Microands Used:

Sequence Number	Description
35	Run multiply, extend Final sign and output through the Large Adder.

Initialization:

- Write multiplier, 0000 0000 0000 0008(HEX), in RF0C.

NOTE

This subsection uses the ALN soft control memory map that was written in section 6, subsection 4.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write multiplicand in RF0B.
a. Report multiplicand; informational.

C.2 Write sign value in RF09.
a. Report Sign Value; informational.

Conditions:

C.3 Test entry of Multiply/Divide final sign.
a. Start processor at Micrand Sequence 35; run multiply, extend final sign, and output through large adder.
b. Wait halt.
c. Read and report contents of RF0A bits 16 through 63.
d. Error if not equal to expected result.

C.4 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.3.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

Iteration Count	Multiplicand	Sign Value	Expected Result
0	0000 0000 0000 0000	0000 0000 0000 0000	0000 FFFF FFFF FF01
1	0000 0000 0000 0001	0000 0000 0000 0000	0000 FFFF FFFF FF03
2	0000 0000 0000 0000	8000 0000 0000 0000	0000 FFFF FFFF FF00
3	0000 0000 0000 0001	8000 0000 0000 0000	0000 FFFF FFFF FF02

Section 6, Subsection 7

This subsection tests Major Cycle Control buffer for holding of U4, U6, and U8.

Hardware Tested:

- Major Cycle Control buffer.

Method:

The testing of the Major Cycle Control buffer for holding of C register controls and 170 format will be performed in the following manner.

The major cycle controls will be in the first micrand issued and zero data will be sent. The second micrand will issue with Multiply/Divide Run, all major cycle controls cleared and a test pattern on ADATC. The third micrand will copy the Multiply/Divide Output mux through the ALN General network.

The ALN soft control entry points used by all micrand sequences in this subsection will enable loading the C register according to the major cycle control, sign extend the C register, shift the C register one time and stop the multiply unit with select the C register to the Multiply/Divide Output mux. This operation is performed twice for each sequence; the first for zero data and the second with the test pattern on ADATC.

Micrands Used:

Sequence Number	Description
36	Test Major Cycle Control buffer for holding of U6.
37	Test Major Cycle Control buffer for holding of U4 and U6.
38	Test Major Cycle Control buffer for holding of U8.

Initialization:

- Write the following ALN soft control memory map (hexadecimal) into the locations specified.

Location	32-bit Value to be Written
0	5108 0100
1	C108 0000
2	0108 0000
3	0000 0000
4	0000 0000
5	0000 0000
6	0000 0000
7	0000 0000

- Write 0000 0000 0000 0000(HEX), in RF0B.

Conditions:

C.0 Write test pattern of 0000 0000 DFFF FFFF in RF0C.
a. Report test pattern; informational.

C.1 Test Major Cycle Control buffer with 32-bit sign extension.
a. Start processor at Micrand Sequence 36; test Major Cycle Control buffer for holding of U6.
b. Wait halt.
c. Read and report contents of RF0A bits 4 through 63.
d. Error if not equal to 0FFF FFFF FFFD FFFF(HEX).

Conditions:

C.2 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.3 Write test pattern of 7AAA 0000 DFFF FFFF(HEX) in RF0C.
a. Report test pattern; informational.

C.4 Test Major Cycle Control buffer with entry of bit 32 extended in bits 16 through 31 and ADATC bits 0 through 15 in C register bits 0 through 15.
a. Start processor at Microand Sequence 37; test Major Cycle Control register for holding of U4 and U8.
b. Wait halt.
c. Read and report contents of RF0A bits 4 through 63.
d. Error if not equal to 0FF7 AAAF FFFD FFFF(HEX).

C.5 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Write test pattern of 4000 7FFF 7FFF 7FFF(HEX) in RF0C.
a. Report test pattern; informational.

C.7 Test Major Cycle Control buffer with 170 sign extension.
a. Start processor at Microand Sequence 38; test Major Cycle Control buffer for holding of U8.
b. Wait halt.
c. Read and report contents of RF0A bits 4 through 63.
d. Error if not equal to 0FFF FFF7 FFF7 FFF7(HEX).

C.8 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.9 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

Section 6, Subsection 8

This subsection tests Major Cycle Control buffer for holding of U1, U2, and U3.

Hardware Tested:

- Major Cycle Control buffer.

Method:

The testing of the Major Cycle Control buffer for holding of B register control bits is performed in the following manner.

The first microand executed will contain the B register controls to be tested and the data sent will be all zeros. The second microand will have all controls clear except Multiply/Divide Run and the test pattern will be sent. It should be entered into the B register according to the major cycle controls of the first microand. The third and fourth microands finish the multiply and output the results through the ALN General network.

The ALN soft control executed by these microands will load the multiplier into the C register, discard the results of the first 12-bit multiply, clock the results of the second 12-bit multiply and output them as MSUM bits 8 through 71 through the Multiply/Divide Output mux. This operation is performed twice; the first time on the zero data and the second time on the test pattern.

Microands Used:

Sequence Number	Description
39	Test Major Cycle Control buffer for U3.
40	Test Major Cycle Control buffer for U2.
41	Test Major Cycle Control buffer for U1.

Initialization:

- Write the following ALN soft control memory map (hexadecimal) into the locations specified.

Location	32-bit Value to be Written
0	4108 0100
1	D308 0000
2	0108 0000
3	0000 0000
4	0114 0000
5	0000 0000
6	0000 0000
7	0000 0000

- Write multiplier, 8000 0000 0000 0800(HEX), in RF0C.
- Write 0000 0000 0000 0000(HEX) in RF09.

Conditions:

C.0 Write test pattern, 0000 0000 DFFF FFFF(HEX) in RF0B.
a. Report test pattern; informational.

C.1 Test Major Cycle Control buffer with 32-bit B register sign extension.
a. Start processor at Microrand Sequence 39; test Major Cycle Control buffer U3.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to FFFF FFFF DFFF FFFF(HEX).

C.2 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.3 Write test pattern, 8000 7FFF 7FFF 7FFF(HEX) in RF0B.
a. Report test pattern; informational.

C.4 Test Major Cycle Control buffer with 48-bit B register sign extension.
a. Start processor at Microrand Sequence 40; test Major Cycle Control buffer for U2.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to FFFF 8000 8000 8001(HEX).

C.5 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Write test pattern, AAAA AAAA AAAA AAAA(HEX) in RF0B.
a. Report test pattern; informational.

C.7 Test Major Cycle Control buffer with 64-bit B register entry.
a. Start processor at Microrand Sequence 41; test Major Cycle Control buffer for U1.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to AAAA AAAA AAAA AAAA(HEX).

C.8 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.9 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

SECTION 7 - VERIFY OPERATION OF THE DIVIDE QUOTIENT REGISTER

This section verifies operation of the Divide Quotient register and its associated logic.

Subsection 0 - Test Divide Quotient register for loading and its output through the C register.

Subsection 1 - Test Divide Quotient register for one's complement and a one place shift.

Subsection 2 - Test Divide Quotient register for two's complement.

Subsection 3 - Test the Major Cycle Hold register for Integer Divide Control, U7.

Subsection 4 - Test output of Divide Quotient register for gating of Divide Fault and Divide Overflow.

Subsection 5 - Test Divide Overflow for select of right shift one in the ALN General Network.

Method:

- Hardware testing: Refer to Subsection Method for details.

Status Reporting:

All conditions in this section that report status will read/format and check all status listed in the Section Status Format except for the multiply carry miscompares.

Initialization:

Initialization is performed as part of subsection 0 PP code.

- Write the following ALN soft control memory map (hexadecimal) into the locations specified.

Location	32-bit Value to be Written
0	7920 0100
1	0300 0000
2	0108 0000
3	0000 0000
4	0308 7920
5	4108 0320
6	0108 0108
7	0000 0000
8	0308 4108
9	0108 0108
A	0000 0000
B	0000 0000

Section Status Format:

The following bit assignments are used by all conditions that report status.

NOTE

Any status not read by a condition and all unused bit positions will be reported as zero. Refer to Section Method for all status that will be read and reported.

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
0	Soft Control Register Parity Error byte 0	PFSALN(62)	PFS88/14
1	Soft Control Register Parity Error byte 1	PFSALN(63)	PFS88/15
2	Not Used		
3	Not Used		
4	Not Used		
5	Not Used		
6	Multiply Carry Miscompare byte 0	PFSALN(44)	PFS88/60
7	Multiply Carry Miscompare byte 1	PFSALN(45)	PFS88/61
8	Not Used		
9	Multiply Carry Miscompare byte 2	PFSALN(46)	PFS89/0
10	Multiply Carry Miscompare byte 3	PFSALN(47)	PFS89/1

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
11	Multiply Carry Miscompare byte 4	PFSALN(48)	PFS89/2
12	Multiply Carry Miscompare byte 5	PFSALN(49)	PFS89/3
13	Multiply Carry Miscompare byte 6	PFSALN(50)	PFS89/4
14	Multiply Carry Miscompare byte 7	PFSALN(51)	PFS89/5
15	Multiply Carry Miscompare byte 8	PFSALN(52)	PFS89/6
16	Multiply C Register Parity Error byte 0	PFSALN(0)	PFS88/16
17	Multiply C Register Parity Error byte 1	PFSALN(1)	PFS88/17
18	Multiply C Register Parity Error byte 2	PFSALN(2)	PFS88/18
19	Multiply C Register Parity Error byte 3	PFSALN(3)	PFS88/19
20	Multiply C Register Parity Error byte 4	PFSALN(4)	PFS88/20
21	Multiply C Register Parity Error byte 5	PFSALN(5)	PFS88/21
22	Multiply C Register Parity Error byte 6	PFSALN(6)	PFS88/22
23	Multiply C Register Parity Error byte 7	PFSALN(7)	PFS88/23
24	Multiply B Register Parity Error byte 0	PFSALN(8)	PFS88/24
25	Multiply B Register Parity Error byte 1	PFSALN(9)	PFS88/25
26	Multiply B Register Parity Error byte 2	PFSALN(10)	PFS88/26
27	Multiply B Register Parity Error byte 3	PFSALN(11)	PFS88/27
28	Multiply B Register Parity Error byte 4	PFSALN(12)	PFS88/28
29	Multiply B Register Parity Error byte 5	PFSALN(13)	PFS88/29
30	Multiply B Register Parity Error byte 6	PFSALN(14)	PFS88/30
31	Multiply B Register Parity Error byte 7	PFSALN(15)	PFS88/31
32	DAI Data Parity Error byte 0	PFSOPI(27)	PFS84/48
33	DAI Data Parity Error byte 1	PFSOPI(28)	PFS84/49
34	DAI Data Parity Error byte 2	PFSOPI(29)	PFS84/50
35	DAI Data Parity Error byte 3	PFSOPI(30)	PFS84/51
36	DAI Data Parity Error byte 4	PFSOPI(31)	PFS84/52
37	DAI Data Parity Error byte 5	PFSOPI(32)	PFS84/53
38	DAI Data Parity Error byte 6	PFSOPI(33)	PFS84/54
39	DAI Data Parity Error byte 7	PFSOPI(34)	PFS84/55

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
40	ALN Micrand Parity Error byte 0	PFSALN(54)	PFS89/8
41	ALN Micrand Parity Error byte 1	PFSALN(55)	PFS89/9
42	ALN Micrand Parity Error byte 2	PFSALN(56)	PFS89/10
43	ALN Micrand Parity Error byte 3	PFSALN(57)	PFS89/11
44	ALN Micrand Parity Error byte 4	PFSALN(58)	PFS89/12
45	ALN Micrand Parity Error byte 5	PFSALN(59)	PFS89/13
46	ALN Micrand Parity Error byte 6	PFSALN(60)	PFS89/14
47	ALN Micrand Parity Error byte 7	PFSALN(61)	PFS89/15
48	Not Used		
49	Not Used		
50	Not Used		
51	Not Used		
52	Processor Did Not Halt Flag from PP Program		
53	Control Store Address bit 0		CSA/53
54	Control Store Address bit 1		CSA/54
55	Control Store Address bit 2		CSA/55
56	Control Store Address bit 3		CSA/56
57	Control Store Address bit 4		CSA/57
58	Control Store Address bit 5		CSA/58
59	Control Store Address bit 6		CSA/59
60	Control Store Address bit 7		CSA/60
61	Control Store Address bit 8		CSA/61
62	Control Store Address bit 9		CSA/62
63	Control Store Address bit 10		CSA/63

Section 7, Subsection 0

This subsection tests Divide Quotient register loading and its output through the C register.

Hardware Tested:

- Divide Arrays Quotient register.
- C register entry from Divide Quotient register.
- Minor cycle control of the Quotient register.

Method:

The testing of entry and output of the Divide Quotient register is performed using the following minor cycle controls.

- Cycle 1 uses V10 to enable clocking of the Quotient register, V3 and V4 to select entry of the ADATB operand, and V1 to select ADATC operand to the C register for parity checking.
- Cycle 2 will strobe B and C register parity.
- Cycle 3 and 4 are effectively no-ops and are used to hold the Divide Quotient register frozen for use by cycle 5.
- Cycle 5 will transfer the Divide Quotient register to C register with a one place left shift.
- Cycle 6, 7, and 8 are effectively no-ops.
- Cycle 9 selects the C register through the Multiply/Divide Output mux.

The major cycle control enters B and C register with unsigned values and disables sign selection to the Divide quotient register. It then copies the Multiply/Divide Output mux through the ALN General network.

All status listed in the Section Status Format except Multiply Carry Miscompares will be read, checked, and reported.

Micrands Used:

Sequence Number	Description
68	Start and Output Divide, unsigned.

Initialization:

- Write 0000 0000 0000 0000(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

Conditions:

C.2 Test Divide Quotient register.
- a. Start processor at Microrand Sequence 68; start and output divide, unsigned.
- b. Wait halt.
- c. Read and report contents of RF0A.
- d. Error if not equal to expected result.

C.3 Read/format and report status.

- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
- a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

Iteration Count	Test Pattern	Expected Result
0	0000 0000 0000 0000	0000 0000 0000 0000
1	5555 5555 5555 5555	AAAA AAAA AAAA AAAA
2	AAAA AAAA AAAA AAAA	5555 5555 5555 5554
3	FFFF FFFF FFFF FFFF	FFFF FFFF FFFF FFFE
4	3737 3737 3737 3737	6E6E 6E6E 6E6E 6E6E

Section 7, Subsection 1

This subsection tests the Divide Quotient register for one's complement and a one place shift.

Hardware Tested:

- Divide Arrays Quotient register.

Method:

The testing of the Divide Quotient register for one's complement and a one place left-shift is done using the following Minor Cycle controls.

- Cycle 1. This cycle will load the Quotient register with ADATB value and selected sign according to V3, V4 and V10. At the same time the B and C registers are loaded so that parity can be checked in the next cycle.
- Cycle 2. This cycle will strobe the Parity Checkers, and V10 without V3 and V4 will load the Quotient register with the contents of the Quotient register left-shifted one place and one's complement if the sign is negative.
- Cycle 3 and 4 are effectively no-ops and hold the Quotient register for use by Cycle 5.
- Cycle 5 transfers the Quotient register to the C register with a one place left-shift.
- Cycle 6, 7, and 8 are effectively no-ops.
- Cycle 9 enables the C register to the Multiply/Divide Output mux. This control is used through the rest of the major cycle.

The major cycle controls used will enter the B and C registers with 64-bit signed values (bit 0 is sign). The selected sign is sent to the Divide Quotient register. The second microrand will transfer the quotient to the C register and the third microrand will transfer the Multiply/Divide Output mux through the ALN General network.

This subsection will read/check and report all status listed in the Section Status Format except for the Multiply Carry Miscompares.

Microrands Used:

Sequence Number	Description
69	Start and output divide, signed.

Initialization:

- Write 0000 0000 0000 0000 in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test quotient register shift.
a. Start processor at Micrand Sequence 69; start and output divide, signed.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

Iteration Count	Test Pattern	Expected Result
0	0000 0000 0000 0000	0000 0000 0000 0000
1	5555 5555 5555 5555	5555 5555 5555 5554
2	2AAA AAAA AAAA AAAA	AAAA AAAA AAAA AAA8
3	7FFF FFFF FFFF FFFF	FFFF FFFF FFFF FFFC
4	3737 3737 3737 3737	DCDC DCDC DCDC DCDC
5	8000 0000 0000 0000	FFFF FFFF FFFF FFFC
6	D555 5555 5555 5555	AAAA AAAA AAAA AAA8
7	AAAA AAAA AAAA AAAA	5555 5555 5555 5554
8	FFFF FFFF FFFF FFFF	0000 0000 0000 0000
9	C8C8 C8C8 C8C8 C8C8	DCDC DCDC DCDC DCDC

Section 7, Subsection 2

This subsection tests the Divide Quotient register for two's complement.

Hardware Tested:

- Divide Quotient register.
- Two's complement Carry Interchange for Divide Quotient register.

Method:

The testing of the Divide Quotient register two's complement logic is performed in the following manner.

Minor Cycle Controls

- Cycle 1. Load Divide Quotient register with ADATB operand.
- Cycle 2. Load two's complemented quotient, left-shifted one place.
- Cycles 3 and 4. Divide Quotient register is held for output by Cycle 5.
- Cycle 5. Enable Divide Quotient register to C register, left-shifted by one place.
- Cycle 6, 7, and 8 are essentially a no operation.
- Cycle 9. Enable C register to Multiply/Divide Output mux and freeze unit in this state.

Major Cycle Controls

- Micrand 1 will cause the Divide Quotient register to be loaded with a 64-bit integer and enable the two's complement.
- Micrand 2 will transfer the quotient to the C register.
- Micrand 3 will output the Multiply/Divide Output mux through the ALN General network.

Status Reporting:

All status listed in the Section Status Format, except the Multiply Carry Miscompares, will be read, checked, and reported.
Micrands Used:

Sequence Number	Description
70	Start and output divide, two's complement integer.

Initialization:

- Write 0000 0000 0000 0000(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test Quotient register two's complement.
a. Start processor at Microand Sequence 70; start and output divide, two's complement integer.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

Iteration Count	Test Pattern	Expected Result
0	0000 0000 0000 0000	0000 0000 0000 0000
1	FFFF FFFF FFFF FFFF	0000 0000 0000 0004
2	FFFF FFFF FFFF FFFE	0000 0000 0000 0008
3	FFFF FFFF FFFF FFFC	0000 0000 0000 0010
4	FFFF FFFF FFFF FFF8	0000 0000 0000 0020
5	FFFF FFFF FFFF FFF0	0000 0000 0000 0040
.	.	.
.	.	.
3B	FC00 0000 0000 0000	1000 0000 0000 0000

Iteration Count	Test Pattern	Expected Result
3C	F800 0000 0000 0000	2000 0000 0000 0000
3D	F000 0000 0000 0000	4000 0000 0000 0000
3E	E000 0000 0000 0000	8000 0000 0000 0000
3F	C000 0000 0000 0000	0000 0000 0000 0000
40	8000 0000 0000 0000	0000 0000 0000 0000

Section 7, Subsection 3

This subsection tests the Major Cycle Hold register for Integer Divide Control U7.

Hardware Tested:

- Major Cycle Hold register.

Method:

The testing of the Major Cycle Hold register for Integer Divide Control, U7, will be performed by executing a microand sequence that has the Multiply/Divide controls including U7 in the first microand. The ALN soft control used will not enable clocking of the Divide arrays. The second microand will have all the Multiply/Divide controls cleared, except Multiply/Divide Run, which causes the major cycle control from the first microand to be used. The ALN soft control used will load the Quotient register, do a two's complement on the quotient, enabled by U7, load the complemented number left-shifted one place into the Quotient register, transfer the Quotient register with a one place left-shift into the C register, and enable the C register to the Multiply/Divide Output mux. The third microand will transfer the quotient to the C register. The fourth microand copies the Multiply/Divide Output mux through the ALN General network.

Microands Used:

Sequence Number	Description
74	Test the Major Cycle Hold Buffer for U7. This sequence uses ALN soft control entry points 11(HEX), 02(HEX), and 12(HEX).

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test U7 in Major Cycle Hold Buffer.
a. Start processor at Micrand Sequence 74; test Major Cycle Hold Buffer for U7.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

Iteration Count	Test Pattern	Expected Result
0	0000 0000 0000 0000	0000 0000 0000 0000
1	FFFF FFFF FFFF FFFF	0000 0000 0000 0004
2	FFFF FFFF FFFF FFFE	0000 0000 0000 0008
3	FFFF FFFF FFFF FFFC	0000 0000 0000 0010
4	FFFF FFFF FFFF FFF8	0000 0000 0000 0020
5	FFFF FFFF FFFF FFF0	0000 0000 0000 0040

Section 7, Subsection 4

This subsection tests outputs of Divide Quotient register for gating of divide fault and divide overflow.

Hardware Tested:

- Divide support logic for detection of divide overflow and divide fault from Quotient register bits.
- ALN exponent aux for enabling of divide error codes and the receipt of the error signals from the divide unit.

Method:

The testing of the Divide Quotient register for gating of divide overflow and divide fault is performed in the following manner:

- Minor cycle control will gate the test pattern into the Divide Quotient register where it will be held for sampling by the Divide Support logic based on the major cycle controls for C register entry and the Integer Divide logic.

- Major cycle control in the first micrand will determine which bits of the Quotient register are sampled for divide fault and overflow. The second micrand will enable the check for and reporting of divide fault. The third micrand will enable the ALN exponent aux to check and report divide overflow. A divide fault should block reporting of divide overflow.

The last micrand will clock the condition registers.

The User Condition register will be read and checked for the proper status.

This subsection will only read, check, and report the following status:

- ALN Micrand Parity Status
- B Register Parity Status
- C Register Parity Status
- Minor Cycle Control Register Parity Status
- Processor Halted Status
- Processor Halt Address

Micrands Used:

Sequence Number	Description
23	Clear UCR.
45	Enable 64-bit integer divide fault and overflow.
46	Enable 32 bit integer divide fault and overflow.
47	Enable 48 bit divide fault and overflow.
48	Read/clear UCR.

Initialization:

- Write 4AAA AAAA 4AAA AAAA(HEX) in RF0C.
- Start processor at Microrand Sequence 23; clear UCR.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test 64-bit divide fault and overflow.
a. Start processor at Microrand Sequence 45; enable 64-bit integer divide fault and overflow.
b. Wait halt.
c. Read/format status.
d. Clear processor to enable write of register file blocked by divide fault.
e. Start processor at Microrand Sequence 48; read/clear UCR.
f. Read and report contents of RF0D bits 55 and 57 through 62.
g. Error if not equal to expected result.

C.3 Report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Test 32-bit divide fault and overflow.
a. Start processor at Microrand Sequence 46; enable 32-bit integer divide fault and overflow.
b. Wait halt.
c. Read/format status.
d. Clear processor to enable write of register file blocked by divide fault.
e. Start processor at Microrand Sequence 48; read/clear UCR.
f. Read and report contents of RF0D bits 55 and 57 through 62.
g. Error if not equal to expected result.

C.5 Report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Test 48-bit divide fault and overflow.
a. Start processor at Microrand Sequence 47; enable 48-bit divide fault and overflow.
b. Wait halt.
c. Read/format status.
d. Clear processor to enable write of register file blocked by divide fault.
e. Start processor at Microrand Sequence 48; read/clear UCR.
f. Read and report contents of RF0D bits 55 and 57 through 62.
g. Error if not equal to expected result.

Conditions:

C.7 Report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Only bits 48 through 55 of the expected result are shown. Bits 0 through 47 will always be zeros.

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 48 through 63
0	0000 0000 0000 0000	C.2	0000
		C.4	0000
		C.6	0000
1	2000 0000 0000 0000	C.2	0040
		C.4	0000
		C.6	0040
2	0400 0000 0000 0000	C.2	0100
		C.4	0000
		C.6	0000
3	0000 0400 0000 0000	C.2	0000
		C.4	0000
		C.6	0100
4	0000 0000 2000 0000	C.2	0000
		C.4	0040
		C.6	0000
5	0000 0000 0400 0000	C.2	0000
		C.4	0100
		C.6	0000

Iteration Count	Test Pattern	Conditions	Expected Result for Bits 48 through 63
6	2400 0000 0000 0000	C.2	0100
		C.4	0000
		C.6	0040
7	0000 0000 2400 0000	C.2	0000
		C.4	0100
		C.6	0000
8	2400 0400 2400 0000	C.2	0100
		C.4	0100
		C.6	0100
9	5BFF FBFF 5BFF FFFF	C.2	0000
		C.4	0000
		C.6	0000
A	8400 0000 8400 0000	C.2	0000
		C.4	0000
		C.6	0000

Section 7, Subsection 5

This subsection tests divide overflow for selecting right-shift one in the ALN General network.

Hardware Tested:

- AND gate that copies divide overflow to MFOVF.

Method:

The testing of the Divide Overflow to MFOVF signal is done by loading Quotient register bit 16 with the first micrand and encoding the Multiply/Divide Floating Overflow with the second micrand.

The second micrand of the sequence causes the following operation in the ALN General network.

- The Shift mux is loaded with RF0B bits 0 through 63 in bits 0 through 63 and RF0C bits 0 through 63 in bits 64 through 127 (Mux Code = 7).
- The Normalize Encoder encodes the MFOVF signal and generates a shift
- The Shifter shifts accordng to the Normalize Encoder.
- The output is taken from the Shift network bits 64 through 127 and written into the register file.

Micrands Used:

Sequence Number	Description
57	Load Quotient and encode MFOVF.

Initialization:

- Write 0248 0000 0000 0000(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test encode of divide overflow.
a. Start processor at Micrand Sequence 57; load quotient and encode MFOVF.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

Iteration Count	Test Pattern	Expected Result
0	0000 0000 0000 0000	0000 0000 0000 0248
1	0000 4000 0000 0000	2000 0000 0000 0124
2	7FFF BFFF FFFF FFFF	BFFF FFFF FFFF 0248

SECTION 8 - CHECKS OF DIVISOR AND REMAINDER REGISTER LOADING

This section verifies the operation of the Divisor and Remainder registers and the Multiply/Divide Output mux select of the Multiply/Divide Test Points.

Subsection 0 - Test Divisor register for 64-bit entry.

Subsection 1 - Test Divisor and Remainder registers for 48-bit entry.

Subsection 2 - Test Divisor register for 32-bit Integer entry.

Subsection 3 - Test Remainder register load from the Quotient register.

Subsection 4 - Test Divisor register complement.

Subsection 5 - Test Remainder register for entry of remainder output.

Method:

Refer to subsection for major cycle control.

Minor Cycle Control

- Integer load ALN soft control, entry points 11(HEX) and 1. This soft control loads the Quotient and Divisor registers, strobes the B and C register parity, shifts the Quotient register, clears the Remainder register, forces Select Subtract, and then enables the Multiply/Divide test points to the Multiply/Divide Output mux.

- Two iteration integer ALN soft control, entry points 10(HEX) and 1. This soft control loads the Divisor and Quotient registers, strobes the B and C register Parity, shifts the Quotient register, clears the Remainder register, forces Select Subtract, enables the Remainder Output to the Remainder register, again forces subtract, and then enables the Multiply/Divide test points to the Multiply/Divide Output mux.

- Floating Point load ALN soft control, entry points 0 and 1. This soft control loads the Divisor and Quotient registers, strobes the B and C register Parity, shifts the Quotient register, enables 47 bits of the Quotient register to the Remainder register, forces Select Subtract, and then enables the Multiply/Divide test points to the Multiply/Divide Output mux.

Status Reporting:

All subsections in this section will read/format and report the following status according to the Section Status Format.

- Soft Control Register Parity Status.
- B Register Parity Status.
- C Register Parity Status.
- Processor Halted Status.
- Processor Halt Address - CSA.

NOTE

The Multiply/Divide test points from the Multiply/Divide Output mux do not have corresponding parity. Random parity errors may be seen in the PFS registers. The status reporting for this section ignores these random errors.

Definitions of terms used in this section:

- Group Propagate — All 4 bit positions of the remainder output are set.
- Group Generate — There is a carry out of the 4-bit group.
- Carry Out — If bit position 23 of the 4-bit group has an enable input (10 binary or 01 binary), and there is a carry into bit position 23, there is a carry out of the 4-bit group.

Initialization:

Initialization is performed as part of subsection 0 PP code.

- Write the following ALN soft control memory map (hexadecimal) into the locations specified.

Location	32-bit Value to be Written
0	7920 7920
1	2320 0320
2	0108 0D20
3	0000 0108
4	0318 7920
5	0118 0320
6	0000 0108
7	0000 0000
8	0318 0000
9	0118 0000
A	0000 0000
B	0000 0000

Section Status Format:

The following bit assignments are used by all conditions that report status.

NOTE

Any status not read by a condition and all unused bit positions are reported as zero. Refer to Section Method for status that will be read and reported.

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
0	Soft Control Register Parity Error byte 0	PFSALN(62)	PFS88/14
1	Soft Control Register Parity Error byte 1	PFSALN(63)	PFS88/15
2	Not Used		
3	Not Used		
4	Not Used		
5	Not Used		
6	Multiply Carry Miscompare byte 0	PFSALN(44)	PFS88/60
7	Multiply Carry Miscompare byte 1	PFSALN(45)	PFS88/61
8	Not Used		
9	Multiply Carry Miscompare byte 2	PFSALN(46)	PFS89/0
10	Multiply Carry Miscompare byte 3	PFSALN(47)	PFS89/1

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
11	Multiply Carry Miscompare byte 4	PFSALN(48)	PFS89/2
12	Multiply Carry Miscompare byte 5	PFSALN(49)	PFS89/3
13	Multiply Carry Miscompare byte 6	PFSALN(50)	PFS89/4
14	Multiply Carry Miscompare byte 7	PFSALN(51)	PFS89/5
15	Multiply Carry Miscompare byte 8	PFSALN(52)	PFS89/6
16	Multiply C Register Parity Error byte 0	PFSALN(0)	PFS88/16
17	Multiply C Register Parity Error byte 1	PFSALN(1)	PFS88/17
18	Multiply C Register Parity Error byte 2	PFSALN(2)	PFS88/18
19	Multiply C Register Parity Error byte 3	PFSALN(3)	PFS88/19
20	Multiply C Register Parity Error byte 4	PFSALN(4)	PFS88/20
21	Multiply C Register Parity Error byte 5	PFSALN(5)	PFS88/21
22	Multiply C Register Parity Error byte 6	PFSALN(6)	PFS88/22
23	Multiply C Register Parity Error byte 7	PFSALN(7)	PFS88/23
24	Multiply B Register Parity Error byte 0	PFSALN(8)	PFS88/24
25	Multiply B Register Parity Error byte 1	PFSALN(9)	PFS88/25
26	Multiply B Register Parity Error byte 2	PFSALN(10)	PFS88/26
27	Multiply B Register Parity Error byte 3	PFSALN(11)	PFS88/27
28	Multiply B Register Parity Error byte 4	PFSALN(12)	PFS88/28
29	Multiply B Register Parity Error byte 5	PFSALN(13)	PFS88/29
30	Multiply B Register Parity Error byte 6	PFSALN(14)	PFS88/30
31	Multiply B Register Parity Error byte 7	PFSALN(15)	PFS88/31
32	DAI Data Parity Error byte 0	PFSOPI(27)	PFS84/48
33	DAI Data Parity Error byte 1	PFSOPI(28)	PFS84/49
34	DAI Data Parity Error byte 2	PFSOPI(29)	PFS84/50
35	DAI Data Parity Error byte 3	PFSOPI(30)	PFS84/51
36	DAI Data Parity Error byte 4	PFSOPI(31)	PFS84/52
37	DAI Data Parity Error byte 5	PFSOPI(32)	PFS84/53
38	DAI Data Parity Error byte 6	PFSOPI(33)	PFS84/54
39	DAI Data Parity Error byte 7	PFSOPI(34)	PFS84/55

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
40	ALN Micrand Parity Error byte 0	PFSALN(54)	PFS89/8
41	ALN Micrand Parity Error byte 1	PFSALN(55)	PFS89/9
42	ALN Micrand Parity Error byte 2	PFSALN(56)	PFS89/10
43	ALN Micrand Parity Error byte 3	PFSALN(57)	PFS89/11
44	ALN Micrand Parity Error byte 4	PFSALN(58)	PFS89/12
45	ALN Micrand Parity Error byte 5	PFSALN(59)	PFS89/13
46	ALN Micrand Parity Error byte 6	PFSALN(60)	PFS89/14
47	ALN Micrand Parity Error byte 7	PFSALN(61)	PFS89/15
48	Not Used		
49	Not Used		
50	Not Used		
51	Not Used		
52	Processor Did Not Halt Flag from PP Program		
53	Control Store Address bit 0		CSA/53
54	Control Store Address bit 1		CSA/54
55	Control Store Address bit 2		CSA/55
56	Control Store Address bit 3		CSA/56
57	Control Store Address bit 4		CSA/57
58	Control Store Address bit 5		CSA/58
59	Control Store Address bit 6		CSA/59
60	Control Store Address bit 7		CSA/60
61	Control Store Address bit 8		CSA/61
62	Control Store Address bit 9		CSA/62
63	Control Store Address bit 10		CSA/63

Section 8, Subsection 0

This subsection tests the Divisor register for 64-bit entry.

Hardware Tested:

- Divide Array logic for entry of divisor, force clear of Remainder register and subtract group propagate and group generate determination.
- Multiply/Divide Output mux for selection and output of Divide Test points.
- Subtract Propagate 1 logic.

Method:

The testing of the Divide Arrays Divisor register, clearing of Remainder register, group propagate and generate determination, and the output of the divide test points is done by setting one bit in the divisor and checking its effect on the group propagate for the 4-bit group that includes that bit. As subtract is always selected for the first iteration, all groups should show group propagate except for the 4-bit groups containing the divisor bit. All group generate bits should be set except for group generate bit 16 when the subtract Propagate 1 carries out of the low order divide array.

The minor cycle control used for this subsection is the integer load ALN soft control described in the Section Method.

The major cycle control enters the operands in the Multiply/Divide unit as 64-bit integers and then output the Multiply/Divide Output mux through the ALN General network.

Micrands Used:

Sequence Number	Description
70	Start and Output Divide, two's complement integer.

Initialization:

- Write Remainder register sensitizing pattern, 0000 FFFF FFFF FFFF(HEX) in RF0B.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern (divisor) in RF0C.
  a. Report test pattern; informational.

C.2 Test Divisor register.
  a. Start processor at Micrand Sequence 70; Start and Output Divide, two's complement integer.
  b. Wait halt.
  c. Read and report contents of RF0A bits 0 through 47.
  d. Error if not equal to expected result.

C.3 Read/format and report status.
  a. Error if not equal to 0000 0000 0000 0003(HEX).

Conditions:

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Only bits 0 through 47 of the expected result are shown; bits 48 through 63 are always zeros.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 47
0	0000 0000 0000 0000	0000 FFFE FFFE
1	0000 0000 0000 0001	0000 FFFE FFFF
2	0000 0000 0000 0002	0000 FFFE FFFF
3	0000 0000 0000 0004	0000 FFFE FFFF
4	0000 0000 0000 0008	0000 FFFE FFFF
5	0000 0000 0000 0010	0000 FFFC FFFE
6	0000 0000 0000 0020	0000 FFFC FFFE
7	0000 0000 0000 0040	0000 FFFC FFFE
8	0000 0000 0000 0080	0000 FFFC FFFE
9	0000 0000 0000 0100	0000 FFFA FFFE
A	0000 0000 0000 0200	0000 FFFA FFFE
B	0000 0000 0000 0400	0000 FFFA FFFE
C	0000 0000 0000 0800	0000 FFFA FFFE

Iteration Count	Test Pattern	Expected Result for Bits 0 through 47
D	0000 0000 0000 1000	0000 FFF8 FFFE
.	.	.
.	.	.
.	.	.
.	.	.
3C	0800 0000 0000 0000	0000 BFFE FFFE
3D	1000 0000 0000 0000	0000 7FFE FFFE
3E	2000 0000 0000 0000	0000 7FFE FFFE
3F	4000 0000 0000 0000	0000 7FFE FFFE

Section 8, Subsection 1

This subsection tests the Divisor and Remainder registers for 48-bit entry.

Hardware Tested:

- Divisor register for entry and zero's extension.
- Remainder register for entry of zeros from Quotient register.
- U1 and U2 control of Divisor register entry.

Method:

The testing of the Divisor and Remainder registers is performed by sending all zeros to the Quotient register for loading in the Remainder register, and by sending the test pattern to the Divisor register.

The test pattern will have bits 1 through 15 always set to check the extend zero's function and will have one other bit set to effect the group propagate for that 4-bit group. The divide group propagates and group generates will be read and checked to verify proper operation of the divide arrays.

The minor cycle control used in this subsection is the floating-point load ALN soft control described in the Section Method.

The major cycle control used will enter the operands in the Multiply/Divide unit as 48-bit sign extended numbers and then output the Multiply/Divide Output mux through the ALN General network.

Microrands Used:

Sequence Number	Description
49	Start and Output Divide, 48-bit entry, signed.

Initialization:

- Write quotient 0000 0000 0000 0000(HEX) in RF0B.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern (divisor) in RF0C.
a. Report test pattern; informational.

C.2 Test Divisor and Remainder register.
a. Start processor at Microrand Sequence 49; Start and Output Divide, 48-bit entry, signed.
b. Wait halt.
c. Read and report contents of RF0A bits 0 through 47.
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Only bits 0 through 47 of the expected result are shown; bits 48 through 63 are always zeros.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 47
0	7FFF 0000 0000 0000	0000 FFFE FFFE
1	7FFF 0000 0000 0001	0000 FFFE FFFF
2	7FFF 0000 0000 0002	0000 FFFE FFFF
3	7FFF 0000 0000 0004	0000 FFFE FFFF
4	7FFF 0000 0000 0008	0000 FFFE FFFF
5	7FFF 0000 0000 0010	0000 FFFC FFFE
6	7FFF 0000 0000 0020	0000 FFFC FFFE
7	7FFF 0000 0000 0040	0000 FFFC FFFE
8	7FFF 0000 0000 0080	0000 FFFC FFFE
9	7FFF 0000 0000 0100	0000 FFFA FFFE
A	7FFF 0000 0000 0200	0000 FFFA FFFE
B	7FFF 0000 0000 0400	0000 FFFA FFFE
C	7FFF 0000 0000 0800	0000 FFFA FFFE
D	7FFF 0000 0000 1000	0000 FFF6 FFFE
.	.	.
.	.	.
.	.	.
.	.	.
.	.	.
2C	7FFF 0800 0000 0000	0000 FBFE FFFE
2D	7FFF 1000 0000 0000	0000 F7FE FFFE
2E	7FFF 2000 0000 0000	0000 F7FE FFFE
2F	7FFF 4000 0000 0000	0000 F7FE FFFE
30	7FFF 8000 0000 0000	0000 F7FE FFFE

Section 8, Subsection 2

This subsection tests the Divisor register for 32-bit integer entry.

Hardware Tested:

- Divisor register in divide arrays.
- U1 and U2 control of divisor entry.

Method:

Testing of the Divisor register for 32-bit integer entry is performed by sending a test pattern to the Divisor register that sensitizes bits 0 through 31, which should be forced to zeros, and set a bit in 33 through 63 to effect the group propagate for that 4-bit group. The Quotient register will be loaded with all ones to sensitize the Remainder register. The divide group propagates and generates will be read and checked to verify proper operation of the divide arrays.

The Minor Cycle control used in this subsection is the integer load ALN soft control described in the Section Method.

The Major Cycle control used in this subsection will enter the operands in the Multiply/Divide unit as 32-bit sign extended integers and output the Multiply/Divide Output mux through the ALN General network.

Microrands Used:

Sequence Number	Description
50	Load and Output Divide, 32-bit signed integer.

Initialization:

- Write Quotient register value 0000 0000 8000 0001(HEX) in RF0B.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern (divisor) in RF0C.
a. Report test pattern; informational.

Conditions:

C.2 Test 32-bit divisor entry.
a. Start processor at Microrand Sequence 50; Load and Output Divide, 32-bit signed integer.
b. Wait halt.
c. Read and report contents of RF0A bits 0 through 47.
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Only bits 0 through 47 of the expected result are shown; bits 48 through 63 are always zeros.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 47
0	FFFF FFFF 0000 0000	0000 FFFE FFFE
1	FFFF FFFF 0000 0001	0000 FFFE FFFF
2	FFFF FFFF 0000 0002	0000 FFFE FFFF
3	FFFF FFFF 0000 0004	0000 FFFE FFFF
4	FFFF FFFF 0000 0008	0000 FFFE FFFF
5	FFFF FFFF 0000 0010	0000 FFFC FFFE
6	FFFF FFFF 0000 0020	0000 FFFC FFFE
7	FFFF FFFF 0000 0040	0000 FFFC FFFE
8	FFFF FFFF 0000 0080	0000 FFFC FFFE
9	FFFF FFFF 0000 0100	0000 FFFA FFFE

Iteration Count	Test Pattern	Expected Result for Bits 0 through 47
A	FFFF FFFF 0000 0200	0000 FFFA FFFE
B	FFFF FFFF 0000 0400	0000 FFFA FFFE
C	FFFF FFFF 0000 0800	0000 FFFA FFFE
D	FFFF FFFF 0000 1000	0000 FFF6 FFFE
.	.	.
.	.	.
.	.	.
.	.	.
.	.	.
1C	FFFF FFFF 0800 0000	0000 FFBE FFFE
1D	FFFF FFFF 1000 0000	0000 FF7E FFFE
1E	FFFF FFFF 2000 0000	0000 FF7E FFFE
1F	FFFF FFFF 4000 0000	0000 FF7E FFFE

Section 8, Subsection 3

This subsection tests the Remainder register load from the Quotient register.

Hardware Tested:

- Divide arrays Remainder register entry.
- U2 control of Remainder register.
- Divide arrays Group Generate logic.
- Interarray Subtract Generate 4 Signals.

Method:

The testing of the Divide Remainder for entry of Quotient register bits 16 through 62 under U2 control is performed by loading the Divisor with all zeros, the Quotient register with the test pattern, and then transferring the Quotient register bits 16 through 62 to the Remainder register bits.

The Automatic Select Subtract on the first iteration will cause the bit set in the test pattern to effect the divide group propagates and group generates. The divide group propagates and group generates are output and checked to verify the proper operation of the divide arrays.

The Minor Cycle control used in this subsection is the floating-point load ALN soft control described in the Section Method.

The Major Cycle control will cause the Multiply/Divide unit to be loaded with a 48-bit sign extended number and output to the Multiply/Divide Output mux.

Microands Used:

Sequence Number	Description
49	Start and Output Divide, 48-bit entry, signed.

Initialization:

- Write divisor 0000 0000 0000 0000(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern quotient in RF0B.
  a. Report test pattern; informational.

C.2 Test Remainder register entry from the Quotient register.
  a. Start processor at Microand Sequence 49; start and output divide, 48-bit entry, signed.
  b. Wait halt.
  c. Read and report contents of RF0A bits 0 through 47.
  d. Error if not equal to expected result.

C.3 Read/format and report status.
  a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
  a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Only bits 0 through 47 of the expected result are shown; bits 48 through 63 are always zeros.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 47
0	7FFF 0000 0000 0000	0000 FFFE FFFE
1	7FFF 0000 0000 0001	0000 FFFE FFFE
2	7FFF 0000 0000 0002	0000 FFFE FFFE
3	7FFF 0000 0000 0004	0000 FFFE FFFE
4	7FFF 0000 0000 0008	0000 FFFE FFFE
5	7FFF 0000 0000 0010	0000 FFFC FFFD
6	7FFF 0000 0000 0020	0000 FFFC FFFC
7	7FFF 0000 0000 0040	0000 FFFC FFFC
8	7FFF 0000 0000 0080	0000 FFFC FFFC
9	7FFF 0000 0000 0100	0000 FFF8 FFFA
A	7FFF 0000 0000 0200	0000 FFFA FFFA
B	7FFF 0000 0000 0400	0000 FFFA FFFA
C	7FFF 0000 0000 0800	0000 FFFA FFFA
D	7FFF 0000 0000 1000	0000 FFF2 FFF6
E	7FFF 0000 0000 2000	0000 FFF6 FFF6
F	7FFF 0000 0000 4000	0000 FFF6 FFF6
10	7FFF 0000 0000 8000	0000 FFF6 FFF6
11	7FFF 0000 0001 0000	0000 FFE6 FFEE
12	7FFF 0000 0002 0000	0000 FFEE FFEE
13	7FFF 0000 0004 0000	0000 FFEE FFEE
14	7FFF 0000 0008 0000	0000 FFEE FFEE
.	.	.
.	.	.
.	.	.
.	.	.
2C	7FFF 0800 0000 0000	0000 FBFE FBFE

Iteration Count	Test Pattern	Expected Result for Bits 0 through 47
2D	7FFF 1000 0000 0000	0000 F3FE F7FE
2E	7FFF 2000 0000 0000	0000 F7FE F7FE
2F	7FFF 4000 0000 0000	0000 F7FE F7FE
30	7FFF 8000 0000 0000	0000 F7FE F7FE

Section 8, Subsection 4

This subsection tests the Divisor register complement.

Hardware Tested:

- Divide Arrays for complement of divisor.
- Integer Divide (U7) for forcing subtract propagate 1 to a zero.

Method:

The testing of the complement of the divisor on a negative number is performed by sending a negative test pattern with all but one bit set to the Divisor register. The negative sign should cause the number to be complemented and the zero input bit in the test pattern will disable the group propagate for that 4-bit group. The negative sign and the Integer Divide control (U7) will disable the subtract propagate 1 into the low order divide array.

The group propagates and group generates will be output and checked to verify proper operation of the Divisor register.

The Minor Cycle control used in this subsection is the integer load ALN soft control described in the Section Method.

The Major Cycle control used in this subsection will do a 64-bit two's complement integer load of the Divisor and Quotient register and then output the Multiply/Divide Output mux through the ALN General network.

Microands Used:

Sequence Number	Description
70	Start and Output Divide, two's complement integer.

Initialization:

- Write Remainder register sensitizing pattern, 0000 FFFF FFFF FFFF(HEX), in RF0B.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern (divisor) in RF0C.
a. Report test pattern; informational.

C.2 Test Divisor register complement.
a. Start processor at Microrand Sequence 70; Start and Output Divide, two's complement integer.
b. Wait halt.
c. Read and report contents of RF0A bits 0 through 47.
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Only bits 0 through 47 of the expected result are shown; bits 48 through 63 are always zeros.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 47
0	FFFF FFFF FFFF FFFF	0000 FFFE FFFE
1	FFFF FFFF FFFF FFFE	0000 FFFE FFFF
2	FFFF FFFF FFFF FFFD	0000 FFFE FFFF
3	FFFF FFFF FFFF FFFB	0000 FFFE FFFF
4	FFFF FFFF FFFF FFF7	0000 FFFE FFFF
5	FFFF FFFF FFFF FFEF	0000 FFFC FFFE
6	FFFF FFFF FFFF FFDF	0000 FFFC FFFE
7	FFFF FFFF FFFF FFBF	0000 FFFC FFFE
8	FFFF FFFF FFFF FF7F	0000 FFFC FFFE
9	FFFF FFFF FFFF FEFF	0000 FFFA FFFE
.	.	.
.	.	.
.	.	.
3B	FBFF FFFF FFFF FFFF	0000 BFFF FFFF
3C	F7FF FFFF FFFF FFFF	0000 BFFF FFFF
3D	EFFF FFFF FFFF FFFF	0000 7FFF FFFF
3E	DFFF FFFF FFFF FFFF	0000 7FFF FFFF
3F	BFFF FFFF FFFF FFFF	0000 7FFF FFFF

Section 8, Subsection 5

This subsection tests the Divide Remainder register for entry of remainder output.

Hardware Tested:

- Divide arrays Remainder register and Internal Subtract logic.

Method:

The testing of the Divide Remainder register for entry of the remainder output of the previous divide iteration is done by setting one bit at a time in the Divisor register and doing a subtract from zeros, clocking the remainder output into the Remainder register and again subtracting the divisor. The divide group propagates and group generates are output and checked to verify the proper operation of the divide arrays.

The Minor Cycle control used in this subsection is the two-iteration integer ALN soft control described in the Section Method.

The Major Cycle control for this subsection will load the Divisor and Quotient registers with 64-bit unsigned integers and output the Multiply/Divide Output mux through the ALN General network.

Microands Used:

Sequence Number	Description
51	Load and Output Divide, 64-bit unsigned integer.

Initialization:

- Write quotient, 0000 0000 0000 0000(HEX), in RF0B.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern divisor in RF0C.
a. Report test pattern; informational.

C.2 Test Remainder register entry.
a. Start processor at Microand Sequence 51; Load and Output Divide, 64-bit unsigned integer.
b. Wait halt.
c. Read and report contents of RF0A bits 0 through 47.
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Post-Subsection Operations:

- Write 0000 0000 0000 0000(HEX) in RF0A to remove any parity errors.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Only bits 0 through 47 of the expected result are shown; bits 48 through 63 are always zeros.

Iteration Count	Test Pattern	Expected Result for Bits 0 through 47
0	0000 0000 0000 0000	0000 FFFE FFFE
1	0000 0000 0000 0001	0000 FFFF FFFF
2	0000 0000 0000 0002	0000 FFFE FFFF
3	0000 0000 0000 0004	0000 FFFE FFFF
4	0000 0000 0000 0008	0000 FFFE FFFF
5	0000 0000 0000 0010	0000 FFFE FFFF
6	0000 0000 0000 0020	0000 FFFC FFFF
7	0000 0000 0000 0040	0000 FFFC FFFF
8	0000 0000 0000 0080	0000 FFFC FFFF
9	0000 0000 0000 0100	0000 FFFC FFFF
A	0000 0000 0000 0200	0000 FFF8 FFFF
B	0000 0000 0000 0400	0000 FFF8 FFFF
C	0000 0000 0000 0800	0000 FFF8 FFFF
D	0000 0000 0000 1000	0000 FFF0 FFFB
E	0000 0000 0000 2000	0000 FFF0 FFFB

Iteration Count	Test Pattern	Expected Result for Bits 0 through 47
F	0000 0000 0000 4000	0000 FFF0 FFFB
10	0000 0000 0000 8000	0000 FFF0 FFFB
11	0000 0000 0001 0000	0000 FFE8 FFFB
12	0000 0000 0002 0000	0000 FFE8 FFFB
13	0000 0000 0004 0000	0000 FFE8 FFFB
14	0000 0000 0008 0000	0000 FFE8 FFFB
15	0000 0000 0010 0000	0000 FFD8 FFFB
16	0000 0000 0020 0000	0000 FFD8 FFFB
.	.	.
.	.	.
.	.	.
3B	0400 0000 0000 0000	0000 BFF8 FFFB
3C	0800 0000 0000 0000	0000 BFF8 FFFB
3D	1000 0000 0000 0000	0000 7FF8 FFFB
3E	2000 0000 0000 0000	0000 7FF8 FFFB
3F	4000 0000 0000 0000	0000 7FF8 FFFB
40	8000 0000 0000 0000	0000 FFFE FFFE

SECTION 9 - CHECK INTERNAL DIVIDE ARRAY LOGIC

This section verifies the operation of the lower 12 Divide arrays and the Divide Carry Interchange.

Subsection 0 - Test Internal Divide Array Logic and Carry Interchange.

Subsection 1 - Test Divide Carry Interchange Array Internal Logic.

Subsection 2 - Test Divide Round Control.

Method:

Refer to subsection for detailed description.

Minor Cycle Control

The ALN Soft Control entry points used in this subsection are combined in various combinations to run 1 to 5 iterations of divide.

The common input control does the following: loads the Divisor and Quotient registers, shifts the Quotient register, gates 47 bits of Quotient to the Remainder register, forces Select Subtract and strobes B and C register parity.

The common run control (1 to 4 copies) does the following: loads the Remainder register with the Remainder Output, shifts the Quotient register, loads the Carry Interchange and allows the Carry Interchange to select Add or Subtract based on the previous iteration results.

The Common Output control enables the Multiply/Divide test points to the Multiply/DivideOutput mux.

Status Reporting:

All subsections in this section will read/format and report the following status according to the Section Status Format.

- Soft Control Register Parity Status.
- B Register Parity Status.
- C Register Parity Status.
- Processor Halted Status.
- Processor Halt Address - CSA.

NOTE

The Multiply/Divide test points from the Multiply/Divide Output mux do not have corresponding parity. Random parity errors may be seen in the PFS registers. The status reporting for this section ignores these random errors.

Definitions of Terms Used in This Section:

- Group Propagate — All 4-bit positions of the remainder output are set.
- Group Generate — There is no carry out of the 4-bit group.
- Carry Out — If bit position 23 of the 4-bit group has an enable input (10 binary or 01 binary), and there is a carry into bit position 23, there is a carry out of the 4-bit group.

Initialization:

Initialization is performed as part of Subsection 0 PP code.

- Write the following ALN Soft Control Memory Map (hexadecimnal) into the locations specified.

Location	32-bit Value to be Written
0	7920 7920
1	2320 2320
2	0108 0DA0
3	0000 0108
4	0318 7920
5	0118 2320
6	0000 0DA0
7	0000 0D60
8	0D20 0D20
9	0108 0D20
A	0000 0108
B	0000 0000

Section Status Format:

The following bit assignments are used by all conditions that report status.

NOTE

Any status not read by a condition and all unused bit positions will be reported as zero. See section method for status that will be read and reported.

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
0	Soft Control Register Parity Error byte 0	PFSALN(62)	PFS88/14
1	Soft Control Register Parity Error byte 1	PFSALN(63)	PFS88/15
2	Not Used		
3	Not Used		
4	Not Used		
5	Not Used		
6	Multiply Carry Miscompare byte 0	PFSALN(44)	PFS88/60
7	Multiply Carry Miscompare byte 1	PFSALN(45)	PFS88/61
8	Not Used		
9	Multiply Carry Miscompare byte 2	PFSALN(46)	PFS89/0
10	Multiply Carry Miscompare byte 3	PFSALN(47)	PFS89/1
11	Multiply Carry Miscompare byte 4	PFSALN(48)	PFS89/2
12	Multiply Carry Miscompare byte 5	PFSALN(49)	PFS89/3
13	Multiply Carry Miscompare byte 6	PFSALN(50)	PFS89/4
14	Multiply Carry Miscompare byte 7	PFSALN(51)	PFS89/5
15	Multiply Carry Miscompare byte 8	PFSALN(52)	PFS89/6
16	Multiply C Register Parity Error byte 0	PFSALN(0)	PFS88/16
17	Multiply C Register Parity Error byte 1	PFSALN(1)	PFS88/17
18	Multiply C Register Parity Error byte 2	PFSALN(2)	PFS88/18
19	Multiply C Register Parity Error byte 3	PFSALN(3)	PFS88/19
20	Multiply C Register Parity Error byte 4	PFSALN(4)	PFS88/20
21	Multiply C Register Parity Error byte 5	PFSALN(5)	PFS88/21
22	Multiply C Register Parity Error byte 6	PFSALN(6)	PFS88/22

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
23	Multiply C Register Parity Error byte 7	PFSALN(7)	PFS88/23
24	Multiply B Register Parity Error byte 0	PFSALN(8)	PFS88/24
25	Multiply B Register Parity Error byte 1	PFSALN(9)	PFS88/25
26	Multiply B Register Parity Error byte 2	PFSALN(10)	PFS88/26
27	Multiply B Register Parity Error byte 3	PFSALN(11)	PFS88/27
28	Multiply B Register Parity Error byte 4	PFSALN(12)	PFS88/28
29	Multiply B Register Parity Error byte 5	PFSALN(13)	PFS88/29
30	Multiply B Register Parity Error byte 6	PFSALN(14)	PFS88/30
31	Multiply B Register Parity Error byte 7	PFSALN(15)	PFS88/31
32	DAI Data Parity Error byte 0	PFSOPI(27)	PFS84/48
33	DAI Data Parity Error byte 1	PFSOPI(28)	PFS84/49
34	DAI Data Parity Error byte 2	PFSOPI(29)	PFS84/50
35	DAI Data Parity Error byte 3	PFSOPI(30)	PFS84/51
36	DAI Data Parity Error byte 4	PFSOPI(31)	PFS84/52
37	DAI Data Parity Error byte 5	PFSOPI(32)	PFS84/53
38	DAI Data Parity Error byte 6	PFSOPI(33)	PFS84/54
39	DAI Data Parity Error byte 7	PFSOPI(34)	PFS84/55
40	ALN Microrand Parity Error byte 0	PFSALN(54)	PFS89/8
41	ALN Microrand Parity Error byte 1	PFSALN(55)	PFS89/9
42	ALN Microrand Parity Error byte 2	PFSALN(56)	PFS89/10
43	ALN Microrand Parity Error byte 3	PFSALN(57)	PFS89/11
44	ALN Microrand Parity Error byte 4	PFSALN(58)	PFS89/12
45	ALN Microrand Parity Error byte 5	PFSALN(59)	PFS89/13
46	ALN Microrand Parity Error byte 6	PFSALN(60)	PFS89/14
47	ALN Microrand Parity Error byte 7	PFSALN(61)	PFS89/15
48	Not Used		
49	Not Used		
50	Not Used		
51	Not Used		
52	Processor Did Not Halt Flag from PP Program		
53	Control Store Address bit 0		CSA/53
54	Control Store Address bit 1		CSA/54

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
55	Control Store Address bit 2		CSA/55
56	Control Store Address bit 3		CSA/56
57	Control Store Address bit 4		CSA/57
58	Control Store Address bit 5		CSA/58
59	Control Store Address bit 6		CSA/59
60	Control Store Address bit 7		CSA/60
61	Control Store Address bit 8		CSA/61
62	Control Store Address bit 9		CSA/62
63	Control Store Address bit 10		CSA/63

Section 9, Subsection 0

This subsection tests Internal Divide Array Logic and Divide Carry Interchange.

Hardware Tested:

- Divide Arrays Internal logic.
- Divide Carry Interchange Array.
- Logic to Select Quotient bit 63 as Division Remainder Least Signficant Bit (BIT).

Method:

Testing of the Divide Array Internal Add/Subtract, Carry Out, and Add of Carry Out to the Remainder logic, along with the Divide Carry Interchange Array for proper selection of the Adder or Subtracter output as the remainder is performed by running one through five iterations of divide on the same test pattern. The divide group propagates and group generates are output and checked after each iteration count to verify proper operation of the Divide logic.

The Minor Cycle control used in this subsection will load the Remainder register with the Quotient register value and then perform the number of divide iterations enabled by the ALN soft control locations used. Refer to the Section Method for details.

The Major Cycle control used in both 2- and 3-microrand sequences, will cause the loading of the Quotient and Divisor values from the test pattern as 48-bit values and then output the Multiply/Divide Output mux through the ALN General network.

Microrands Used:

Sequence Number	Description
49	Start and Output Divide, 48-bit signed.
52	Start and Output Divide, 48-bit signed, from ALN soft control entry points 10(HEX) and 1.
53	Start and Output Divide, 48-bit signed, from ALN soft control entry points 11(HEX) and 1.
54	Start, Run and Output Divide, 48-bit signed, from ALN soft control entry points 11(HEX), 2, and 1.
55	Start, Run and Output Divide, 48-bit signed, from ALN soft control entry points 11(HEX), 12(HEX), and 1.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write quotient from test pattern in RF0B.
- a. Report test pattern quotient; informational.

C.2 Write divisor from test pattern in RF0C.
- a. Report test pattern divisor; informational.

C.3 Test Divide logic with one iteration.
- a. Start processor at Microrand Sequence 49; start and output divide, 48-bit signed.
- b. Wait halt.
- c. Read and report contents of RF0A, bits 0 through 47.
- d. Error if not equal to expected result.

C.4 Read/format and report status.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test Divide logic with two iterations.
- a. Start processor at Microrand Sequence 52; start and output divide, 48-bit signed from ALN soft control entry points 10(HEX) and 1.
- b. Wait halt.
- c. Read and report contents of RF0A, bits 0 through 47.
- d. Error if not equal to expected result.

C.6 Read/format and report status.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

Conditions:

C.7 Test Divide logic with three iterations.
- a. Start processor at Microrand Sequence 53; start and output divide, 48-bit signed from ALN soft control entry points 11(HEX) and 1.
- b. Wait halt.
- c. Read and report contents of RF0A, bits 0 through 47.
- d. Error if not equal to expected result.

C.8 Read/format and report status.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.9 Test Divide logic with four iterations.
- a. Start processor at Microrand Sequence 54; start, run, and output divide, 48-bit signed from ALN soft control entry points 11(HEX), 2, and 1.
- b. Wait halt.
- c. Read and report contents of RF0A, bits 0 through 47.
- d. Error if not equal to expected result.

C.10 Read/format and report status.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Test Divide logic with five iterations.
- a. Start processor at Microrand Sequence 55; start, run, and output divide, 48-bit signed from ALN soft control entry points 11(HEX), 12(HEX), and 1.
- b. Wait halt.
- c. Read and report contents of RF0A, bits 0 through 47.
- d. Error if not equal to expected result.

C.12 Read/format and report status.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.13 Report First Failure parity information.
- a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.3.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Only bits 16 through 47 of the expected result are shown; bits 0 through 15 and 48 through 63 are always zeros.

Iteration Count	Test Pattern for Quotient	Test Pattern for Divisor	Conditions	Expected Result
0	0000 1818 1818 1818	0000 1818 1818 1818	C.3	F000 FAAA
			C.5	FFFC FFFD
			C.7	F550 FAAF
			C.9	F000 FFFF
			C.11	F000 FFFF
1	0000 8181 8181 8181	0000 8181 8181 8181	C.3	F001 FD55
			C.5	FFFE FFFE
			C.7	EAA9 F557
			C.9	F001 FFFF
			C.11	F001 FFFF
2	0000 1414 1414 1414	0000 1818 1818 1818	C.3	F000 FAAA
			C.5	FAAA FFFF
			C.7	F554 F555
			C.9	F000 F555
			C.11	FAAA FFFF
3	0000 4141 4141 4141	0000 8181 8181 8181	C.3	F001 FD55
			C.5	F554 FFFE
			C.7	FAA8 FAAB
			C.9	F000 FAAA
			C.11	F000 FFFF
4	0000 1111 1111 1111	0000 1818 1818 1818	C.3	F000 FAAA
			C.5	FAAA FFFF
			C.7	F000 F555
			C.9	FAAA FFFF
			C.11	F554 F555
5	0000 1111 1111 1111	0000 8181 8181 8181	C.3	F000 FD55
			C.5	F554 FFFE
			C.7	F000 FAAB
			C.9	F000 FFFF
			C.11	E000 EFFF
6	0000 1414 1414 1414	0000 1414 1414 1414	C.3	FAAA FFFF
			C.5	FFFE FFFE
			C.7	F000 FFFF
			C.9	F000 FFFF
			C.11	F000 FFFF
7	0000 4141 4141 4141	0000 4141 4141 4141	C.3	F555 FFFF
			C.5	FFFE FFFE
			C.7	F001 FFFF
			C.9	F001 FFFF
			C.11	F001 FFFF

Iteration Count	Test Pattern for Quotient	Test Pattern for Divisor	Conditions	Expected Result
8	0000 0C0C 0C0C 0C0C	0000 1414 1414 1414	C.3	F000 FAAA
			C.5	FAAA FFFF
			C.7	FAA8 FAA9
			C.9	F00A FAAF
			C.11	FAAA FFFF
9	0000 40C0 C0C0 C0C0	0000 4141 4141 4141	C.3	F001 FD55
			C.5	FD55 FFFF
			C.7	F555 F555
			C.9	F000 F555
			C.11	F000 F555
A	0000 0A0A 0A0A 0A0A	0000 1414 1414 1414	C.3	F000 FAAA
			C.5	FAAA FFFF
			C.7	FFFE FFFE
			C.9	F000 FFFF
			C.11	F000 FFFF
B	0000 20A0 A0A0 A0A0	0000 4141 4141 4141	C.3	F001 FD55
			C.5	F555 FFFF
			C.7	FFFF FFFF
			C.9	F001 F001
			C.11	E000 EAA9
C	0000 0909 0909 0909	0000 1414 1414 1414	C.3	F000 FAAA
			C.5	FAAA FFFF
			C.7	FAAA FFFF
			C.9	F554 F554
			C.11	F000 F555
D	0000 1090 9090 9090	0000 4141 4141 4141	C.3	F000 FD55
			C.5	F555 FFFF
			C.7	F555 FFFF
			C.9	FAAB FAAB
			C.11	F000 FAAB
E	0000 0888 8888 8888	0000 1414 1414 1414	C.3	F554 FD54
			C.5	F800 FFFF
			C.7	FAAA FFFF
			C.9	F000 F555
			C.11	F000 F555
F	0000 0888 8888 8888	0000 4141 4141 4141	C.3	F2AA FAAA
			C.5	F000 FFFF
			C.7	F555 FFFF
			C.9	F001 FAAB
			C.11	F003 F557

Iteration Count	Test Pattern for Quotient	Test Pattern for Divisor	Conditions	Expected Result
10	0000 1212 1212 1212	0000 1212 1212 1212	C.3	FAAA FFFF
			C.5	FFFE FFFE
			C.7	F000 FFFF
			C.9	F000 FFFF
			C.11	F000 FFFF
11	0000 2121 2121 2121	0000 2121 2121 2121	C.3	F555 FFFF
			C.5	FFFE FFFE
			C.7	F001 FFFF
			C.9	F001 FFFF
			C.11	F001 FFFF
12	0000 0606 0606 0606	0000 1212 1212 1212	C.3	F000 FAAA
			C.5	FAAA FFFF
			C.7	FAAA FFFF
			C.9	F000 F555
			C.11	FAAA FFFF
13	0000 2060 6060 6060	0000 2121 2121 2121	C.3	F001 FD55
			C.5	FD55 FFFF
			C.7	F555 F7FF
			C.9	F000 F555
			C.11	F000 F555
14	0000 0505 0505 0505	0000 1212 1212 1212	C.3	F000 FAAA
			C.5	FAAA FFFF
			C.7	FAAA FFFF
			C.9	FAA8 FAA9
			C.11	F000 FAAA
15	0000 1050 5050 5050	0000 2121 2121 2121	C.3	F000 FD55
			C.5	F555 FFFF
			C.7	FD55 FFFF
			C.9	F555 F555
			C.11	F000 F555
16	0000 0484 8484 8484	0000 1212 1212 1212	C.3	F554 FD54
			C.5	F800 FFFF
			C.7	FAAA FFFF
			C.9	FFFE FFFF
			C.11	F000 F001
17	0000 0848 4848 4848	0000 2121 2121 2121	C.3	F2AA FAAA
			C.5	F000 FFFF
			C.7	F555 FFFF
			C.9	FFFF FFFF
			C.11	F001 F001

Iteration Count	Test Pattern for Quotient	Test Pattern for Divisor	Conditions	Expected Result
18	0000 1111 1111 1111	0000 1111 1111 1111	C.3	F800 FFFF
			C.5	FFFE FFFE
			C.7	F001 FFFF
			C.9	F001 FFFF
			C.11	F001 FFFF
19	0000 0202 0202 0202	0000 1111 1111 1111	C.3	F554 FFFE
			C.5	F002 FAAB
			C.7	FAAA FFFF
			C.9	FAAA FFFF
			C.11	FAAB FFFF
1A	0000 0020 2020 2020	0000 1111 1111 1111	C.3	F2AB FFFF
			C.5	F001 FD55
			C.7	F555 FFFF
			C.9	F555 FFFF
			C.11	F555 FFFF
1B	0000 0303 0303 0303	0000 1111 1111 1111	C.3	F000 FAAA
			C.5	FAAA FFFF
			C.7	FAAA FFFF
			C.9	FAAA FFFF
			C.11	F000 F555
1C	0000 1030 3030 3030	0000 1111 1111 1111	C.3	F800 FD55
			C.5	FD55 FFFF
			C.7	F555 F7FF
			C.9	F000 F7FF
			C.11	F000 F555
1D	0000 3838 3838 3838	0000 3838 3838 3838	C.3	F000 FAAA
			C.5	FFFC FFFD
			C.7	F550 FAAF
			C.9	F000 FFFF
			C.11	F000 FFFF
1E	0000 8383 8383 8383	0000 8383 8383 8383	C.3	F000 FD55
			C.5	FFFE FFFE
			C.7	EAA8 F557
			C.9	F000 FFFF
			C.11	F000 FFFF
1F	0000 7878 7878 7878	0000 7878 7878 7878	C.3	F000 FAAA
			C.5	FFFC FFFD
			C.7	F550 FAAF
			C.9	F000 FFFF
			C.11	F000 FFFF

Iteration Count	Test Pattern for Quotient	Test Pattern for Divisor	Conditions	Expected Result
20	0000 8787 8787 8787	0000 8787 8787 8787	C.3	F000 FD55
			C.5	FFFE FFFE
			C.7	EAA8 F557
			C.9	F000 FFFF
			C.11	F000 FFFF
21	0000 F8F8 F8F8 F8F8	0000 F8F8 F8F8 F8F8	C.3	F000 FAAA
			C.5	FFFC FFFD
			C.7	EAA0 FAA7
			C.9	E000 F555
			C.11	E000 F555
22	0000 8F8F 8F8F 8F8F	0000 8F8F 8F8F 8F8F	C.3	F000 FD55
			C.5	FFFE FFFE
			C.7	E550 F553
			C.9	F000 FAAA
			C.11	F000 FAAA
23	0000 1212 1212 1212	0000 1818 1818 1818	C.3	F000 FAAA
			C.5	FAAA FFFF
			C.7	F000 F555
			C.9	FFFE FFFE
			C.11	F554 FAAB
24	0000 2121 2121 2121	0000 8181 8181 8181	C.3	F001 FD55
			C.5	F554 FFFE
			C.7	F000 FAAB
			C.9	E000 EFFF
			C.11	F000 FAAA
25	0000 FFFF FFFF FFFF	0000 FFFF FFFF FFFF	C.3	F7FE FFFE
			C.5	FFFE FFFE
			C.7	EFF8 FFFB
			C.9	EFFE FFFE
			C.11	EFF8 FFFE

Section 9, Subsection 1

This subsection tests Divide Carry Interchange Array Internal logic.

Hardware Tested:

- Divide Carry Interchange.

Method:

The testing of the Divide Carry Interchange Internal logic for propagation of group generates through the group propagates, and no group propagate disable of group generates is performed in the following manner.

The test patterns are structured in such a way that only one group generate will be inactive at a time. If the group propagates above it are all active, the not group generate should propagate through them and select subtract on the second iteration. If any group propagate above the not group generate is inactive, it should cause the second iteration to select the Add Output. The test patterns will be run twice. The first time checks the group propagate and group generate calculation for the first subtract. Then a two-iteration divide checks that the Divide Carry Interchange handled the group propagates and group generates properly.

The minor cycle control used by this subsection is the one- and two-iteration floating-point soft control described in the Section Method.

The major cycle control used by this subsection will cause the loading of the Quotient and Divisor Values, from the test pattern, as 48-bit values, and then output the Multiply/Divide Output mux through the ALN General network.

Microands Used:

Sequence Number	Description
49	Start and output divide, 48-bit signed.
52	Start and output divide, 48-bit signed, from ALN soft control entry points 10(HEX) and 1.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write quotient from test pattern in RF0B.
- a. Report test pattern quotient; informational.

C.2 Write divisor from test pattern in RF0C.
- a. Report test pattern divisor; informational.

C.3 Check group propagates and generates.
- a. Start processor at Microand Sequence 49; start and output divide, 48-bit signed.
- b. Wait halt.
- c. Read and report contents of RF0A, bits 0 through 47.
- d. Error if not equal to expected result.

Conditions:

C.4 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Test Divide Carry Interchange.
a. Start processor at Micrand Sequence 52; start and output divide, 48-bit signed from ALN soft control entry points 10(HEX) and 1.
b. Wait halt.
c. Read and report contents of RF0A, bits 0 through 47.
d. Error if not equal to expected result.

C.6 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.7 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.3.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

NOTE

Only bits 16 through 47 of expected result are shown; bits 0 through 15 and 48 through 63 are always zeros.

Iteration Count	Test Pattern for Quotient	Test Pattern for Divisor	Conditions	Expected Result
0	0000 0000 0000 0000	0000 0000 0000 0000	C.3	FFFE FFFE
			C.5	FFFE FFFE
1	0000 0000 0000 0020	0000 0000 0000 0001	C.3	FFFD FFFD
			C.5	FFFC FFFD
2	0000 0000 0000 0200	0000 0000 0000 0001	C.3	FFFB FFFB
			C.5	FFFA FFFB
3	0000 0000 0000 2000	0000 0000 0000 0001	C.3	FFF7 FFF7
			C.5	FFF6 FFF7
4	0000 0000 0002 0000	0000 0000 0000 0001	C.3	FFEF FFEF
			C.5	FFEE FFEF

Iteration Count	Test Pattern for Quotient	Test Pattern for Divisor	Conditions	Expected Result
5	0000 0000 0020 0000	0000 0000 0000 0001	C.3	FFDF FFDF
			C.5	FFDE FFDF
6	0000 0000 0200 0000	0000 0000 0000 0001	C.3	FFBF FFBF
			C.5	FFBE FFBF
7	0000 0000 2000 0000	0000 0000 0000 0001	C.3	FF7F FF7F
			C.5	FF7E FF7F
8	0000 0002 0000 0000	0000 0000 0000 0001	C.3	FEFF FEFF
			C.5	FEFE FEFF
9	0000 0020 0000 0000	0000 0000 0000 0001	C.3	FDFF FDFF
			C.5	FDFE FDFF
A	0000 0200 0000 0000	0000 0000 0000 0001	C.3	FBFF FBFF
			C.5	FBFE FBFF
B	0000 2000 0000 0000	0000 0000 0000 0001	C.3	F7FF F7FF
			C.5	F7FE F7FF
C	0000 0000 0000 0002	0000 8000 0000 0001	C.3	F7FE FFFE
			C.5	F7F8 FFFB
D	0000 0000 0000 0020	0000 8000 0000 0001	C.3	F7FD FFFD
			C.5	F7F1 FFF7
E	0000 0000 0000 0200	0000 8000 0000 0001	C.3	F7FB FFFB
			C.5	F7E3 FFEF
F	0000 0000 0000 2000	0000 8000 0000 0001	C.3	F7F7 FFF7
			C.5	F7C7 FFDF
10	0000 0000 0002 0000	0000 8000 0000 0001	C.3	F7EF FFEF
			C.5	F78F FFBF
11	0000 0000 0020 0000	0000 8000 0000 0001	C.3	F7DF FFDF
			C.5	F71F FF7F
12	0000 0000 0200 0000	0000 8000 0000 0001	C.3	F7BF FFBF
			C.5	F63F FEFF
13	0000 0000 2000 0000	0000 8000 0000 0001	C.3	F77F FF7F
			C.5	F47F FDFF
14	0000 0002 0000 0000	0000 8000 0000 0001	C.3	F6FF FEFF
			C.5	F0FF FBFF
15	0000 0020 0000 0000	0000 8000 0000 0001	C.3	F5FF FDFF
			C.5	F1FF FFFF

Iteration Count	Test Pattern for Quotient	Test Pattern for Divisor	Conditions	Expected Result
16	0000 0200 0000 0000	0000 8000 0000 0001	C.3	F3FF FBFF
			C.5	F3FF FFFF
17	0000 0000 0000 0002	0000 0800 0000 0001	C.3	FBFE FFFE
			C.5	FBF8 FFFB
18	0000 0000 0000 0020	0000 0800 0000 0001	C.3	FBFD FBFD
			C.5	FBF1 FFF7
19	0000 0000 0000 0200	0000 0800 0000 0001	C.3	FBFB FFFB
			C.5	FBE3 FFEF
1A	0000 0000 0000 2000	0000 0800 0000 0001	C.3	FBF7 FFF7
			C.5	FBC7 FFDF
1B	0000 0000 0002 0000	0000 0800 0000 0001	C.3	FBEF FFEF
			C.5	FB8F FFBF
1C	0000 0000 0020 0000	0000 0800 0000 0001	C.3	FBDF FFDF
			C.5	FB1F FF7F
1D	0000 0000 0200 0000	0000 0800 0000 0001	C.3	FBBF FFBF
			C.5	FA3F FEFF
1E	0000 0000 2000 0000	0000 0800 0000 0001	C.3	FB7F FF7F
			C.5	F87F FDFF
1F	0000 0002 0000 0000	0000 0800 0000 0001	C.3	FAFF FEFF
			C.5	F8FF FFFF
20	0000 0020 0000 0000	0000 0800 0000 0001	C.3	F9FF FDFF
			C.5	F9FF FFFF
21	0000 0000 0000 0002	0000 0080 0000 0001	C.3	FDFE FFFE
			C.5	FDF8 FFFB
22	0000 0000 0000 0020	0000 0080 0000 0001	C.3	FDFD FFFD
			C.5	FDF1 FFF7
23	0000 0000 0000 0200	0000 0080 0000 0001	C.3	FDFB FFFB
			C.5	FDE3 FFEF
24	0000 0000 0000 2000	0000 0080 0000 0001	C.3	FDF7 FFF7
			C.5	FDC7 FFDF
25	0000 0000 0002 0000	0000 0080 0000 0001	C.3	FDEF FFEF
			C.5	FD8F FFBF
26	0000 0000 0020 0000	0000 0080 0000 0001	C.3	FDDF FFDF
			C.5	FD1F FF7F

Iteration Count	Test Pattern for Quotient	Test Pattern for Divisor	Conditions	Expected Result
27	0000 0000 0200 0000	0000 0080 0000 0001	C.3	FDBF FFBF
			C.5	FC3F FEFF
28	0000 0000 2000 0000	0000 0080 0000 0001	C.3	FD7F FF7F
			C.5	FC7F FFFF
29	0000 0002 0000 0000	0000 0080 0000 0001	C.3	FCFF FEFF
			C.5	FCFF FFFF
2A	0000 0000 0000 0002	0000 0008 0000 0001	C.3	FEFE FFFE
			C.5	FEF8 FFFB
2B	0000 0000 0000 0020	0000 0008 0000 0001	C.3	FEFD FFFD
			C.5	FEF1 FFF7
2C	0000 0000 0000 0200	0000 0008 0000 0001	C.3	FEFB FFFB
			C.5	FEE3 FFEF
2D	0000 0000 0000 2000	0000 0008 0000 0001	C.3	FEF7 FFF7
			C.5	FEC7 FFDF
2E	0000 0000 0002 0000	0000 0008 0000 0001	C.3	FEEF FFEF
			C.5	FE8F FFBF
2F	0000 0000 0020 0000	0000 0008 0000 0001	C.3	FEDF FFDF
			C.5	FE1F FF7F
30	0000 0000 0200 0000	0000 0008 0000 0001	C.3	FEBF FFBF
			C.5	FE3F FFFF
31	0000 0000 2000 0000	0000 0008 0000 0001	C.3	FE7F FF7F
			C.5	FE7F FFFF
32	0000 0000 0000 0002	0000 0000 8000 0001	C.3	FF7E FFFE
			C.5	FF78 FFFB
33	0000 0000 0000 0020	0000 0000 8000 0001	C.3	FF7D FFFD
			C.5	FF71 FFF7
34	0000 0000 0000 0200	0000 0000 8000 0001	C.3	FF7B FFFB
			C.5	FF63 FFEF
35	0000 0000 0000 2000	0000 0000 8000 0001	C.3	FF77 FFF7
			C.5	FF47 FFDF
36	0000 0000 0002 0000	0000 0000 8000 0001	C.3	FF6F FFEF
			C.5	FF0F FFBF
37	0000 0000 0020 0000	0000 0000 8000 0001	C.3	FF5F FFDF
			C.5	FF1F FFFF

Iteration Count	Test Pattern for Quotient	Test Pattern for Divisor	Conditions	Expected Result
38	0000 0000 0200 0000	0000 0000 8000 0001	C.3	FF3F FFBF
			C.5	FF3F FFFF
39	0000 0000 0000 0002	0000 0000 0800 0001	C.3	FFBE FFFE
			C.5	FFB8 FFFB
3A	0000 0000 0000 0020	0000 0000 0800 0001	C.3	FFBD FFFD
			C.5	FFB1 FFF7
3B	0000 0000 0000 0200	0000 0000 0800 0001	C.3	FFBB FFFB
			C.5	FFA3 FFEF
3C	0000 0000 0000 2000	0000 0000 0800 0001	C.3	FFB7 FFF7
			C.5	FF87 FFDF
3D	0000 0000 0002 0000	0000 0000 0800 0001	C.3	FFAF FFEF
			C.5	FF8F FFFF
3E	0000 0000 0020 0000	0000 0000 0800 0001	C.3	FF9F FFDF
			C.5	FF9F FFFF
3F	0000 0000 0000 0002	0000 0000 0080 0001	C.3	FFDE FFFE
			C.5	FFD8 FFFB
40	0000 0000 0000 0020	0000 0000 0080 0001	C.3	FFDD FFFD
			C.5	FFD1 FFF7
41	0000 0000 0000 0200	0000 0000 0080 0001	C.3	FFDB FFFB
			C.5	FFC3 FFEF
42	0000 0000 0000 2000	0000 0000 0080 0001	C.3	FFD7 FFF7
			C.5	FFC7 FFFF
43	0000 0000 0002 0000	0000 0000 0080 0001	C.3	FFCF FFEF
			C.5	FFCF FFFF
44	0000 0000 0000 0002	0000 0000 0008 0001	C.3	FFEE FFFE
			C.5	FFE8 FFFB
45	0000 0000 0000 0020	0000 0000 0008 0001	C.3	FFED FFFD
			C.5	FFE1 FFF7
46	0000 0000 0000 0200	0000 0000 0008 0001	C.3	FFEB FFFB
			C.5	FFE3 FFFF
47	0000 0000 0000 2000	0000 0000 0008 0001	C.3	FFE7 FFF7
			C.5	FFE7 FFFF
48	0000 0000 0000 0002	0000 0000 0000 8001	C.3	FFF6 FFFE
			C.5	FFF0 FFFB

Iteration Count	Test Pattern for Quotient	Test Pattern for Divisor	Conditions	Expected Result
49	0000 0000 0000 0020	0000 0000 0000 8001	C.3	FFF5 FFFD
			C.5	FFF1 FFFF
4A	0000 0000 0000 0200	0000 0000 0000 8001	C.3	FFF3 FFFB
			C.5	FFF3 FFFF
4B	0000 0000 0000 0002	0000 0000 0000 0801	C.3	FFFA FFFE
			C.5	FFF8 FFFF
4C	0000 0000 0000 0020	0000 0000 0000 0801	C.3	FFF9 FFFD
			C.5	FFF9 FFFF
4D	0000 0000 0000 0002	0000 0000 0000 0081	C.3	FFFC FFFE
			C.5	FFFC FFFF

Section 9, Subsection 2

This subsection tests divide round control.

Hardware Tested:

- Gate from V9 and U9.
- Division Remainder Least Significant Bit select logic.

Method:

The testing of the Divide Round entry to the Division Remainder least significant bit signal (BIT) is performed by running three iterations of divide with U9 set in the major cycle control and V9 set in the last soft control location that clocks the divide arrays. This causes BIT to set and be entered into the lower position of the Remainder register. The Multiply/Divide test points are read and checked to verify proper operation of the BIT Logic.

Microrands Used:

Sequence Number	Description
56	Test divide round control from ALN soft control entry points 11(HEX) and 1.
66	Test Major Cycle Control buffer for hold of U9.

Initialization:

- Write 0000 0000 0000 0000(HEX) in RF0B.
- Write 0000 8000 0000 0001(HEX) in RF0C.

Conditions:

C.0 Test round control.
a. Start processor at Microrand Sequence 56; test divide round control from ALN soft control entry points 11(HEX) and 1.
b. Wait halt.
c. Read and report the contents of RF0A bits 0 through 47.
d. Error if not equal to 0000 F7FE FFFE 0000(HEX).

C.1 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.2 Test Major Cycle Control buffer for hold of U9.
a. Start processor at Microrand Sequence 66; test Major Cycle Control buffer for hold of U9.
b. Wait halt.
c. Read and report the contents of RF0A bits 10 through 47.
d. Error if not equal to 0000 F7FE FFFE 0000(HEX).

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

Post-Subsection Operations:

- Write 0000 0000 0000 0000(HEX) in RF0A to remove any parity errors.

SECTION 10 - CHECK INTEGER DIVIDE OPERATION AND POP COUNTER

This section verifies the operation of the Internal Array Logic of the upper five Divide arrays, the Divide Carry Interchange, and the Pop Counter.

Subsection 0 - Test Integer entry to the Divide Remainder register.

Subsection 1 - Test upper Divide arrays.

Subsection 2 - Test Pop Counter.

Method:

- Refer to subsection for detailed description.

Status Reporting:

All status listed in the Section Status Format, except Multiply Carry Miscompare, will be read/formatted and checked.

Initialization:

Initialization is performed as part of Subsection 0 PP code.

- Write the following ALN Soft Control Memory Map (hexadecimal) into the locations specified.

Location	32-bit Value to be Written
0	7920 5120
1	0320 5320
2	0D20 0300
3	0D20 0100
4	0D60 C108
5	0D20 010C
6	0D20 0000
7	0D20 0000
8	0308 4108
9	0108 0108
A	0000 0000
B	0000 0000

Section Status Format:

The following bit assignments are used by all conditions that report status.

NOTE

Any status not read by a condition and all unused bit positions will be reported as zero. See section method for status that will be read and reported.

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
0	Soft Control Register Parity Error byte 0	PFSALN(62)	PFS88/14
1	Soft Control Register Parity Error byte 1	PFSALN(63)	PFS88/15
2	Not Used		
3	Not Used		
4	Not Used		
5	Not Used		
6	Multiply Carry Miscompare byte 0	PFSALN(44)	PFS88/60
7	Multiply Carry Miscompare byte 1	PFSALN(45)	PFS88/61
8	Not Used		
9	Multiply Carry Miscompare byte 2	PFSALN(46)	PFS89/0
10	Multiply Carry Miscompare byte 3	PFSALN(47)	PFS89/1
11	Multiply Carry Miscompare byte 4	PFSALN(48)	PFS89/2
12	Multiply Carry Miscompare byte 5	PFSALN(49)	PFS89/3
13	Multiply Carry Miscompare byte 6	PFSALN(50)	PFS89/4
14	Multiply Carry Miscompare byte 7	PFSALN(51)	PFS89/5
15	Multiply Carry Miscompare byte 8	PFSALN(52)	PFS89/6
16	Multiply C Register Parity Error byte 0	PFSALN(0)	PFS88/16
17	Multiply C Register Parity Error byte 1	PFSALN(1)	PFS88/17
18	Multiply C Register Parity Error byte 2	PFSALN(2)	PFS88/18
19	Multiply C Register Parity Error byte 3	PFSALN(3)	PFS88/19
20	Multiply C Register Parity Error byte 4	PFSALN(4)	PFS88/20
21	Multiply C Register Parity Error byte 5	PFSALN(5)	PFS88/21

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
22	Multiply C Register Parity Error byte 6	PFSALN(6)	PFS88/22
23	Multiply C Register Parity Error byte 7	PFSALN(7)	PFS88/23
24	Multiply B Register Parity Error byte 0	PFSALN(8)	PFS88/24
25	Multiply B Register Parity Error byte 1	PFSALN(9)	PFS88/25
26	Multiply B Register Parity Error byte 2	PFSALN(10)	PFS88/26
27	Multiply B Register Parity Error byte 3	PFSALN(11)	PFS88/27
28	Multiply B Register Parity Error byte 4	PFSALN(12)	PFS88/28
29	Multiply B Register Parity Error byte 5	PFSALN(13)	PFS88/29
30	Multiply B Register Parity Error byte 6	PFSALN(14)	PFS88/30
31	Multiply B Register Parity Error byte 7	PFSALN(15)	PFS88/31
32	DAI Data Parity Error byte 0	PFSOPI(27)	PFS84/48
33	DAI Data Parity Error byte 1	PFSOPI(28)	PFS84/49
34	DAI Data Parity Error byte 2	PFSOPI(29)	PFS84/50
35	DAI Data Parity Error byte 3	PFSOPI(30)	PFS84/51
36	DAI Data Parity Error byte 4	PFSOPI(31)	PFS84/52
37	DAI Data Parity Error byte 5	PFSOPI(32)	PFS84/53
38	DAI Data Parity Error byte 6	PFSOPI(33)	PFS84/54
39	DAI Data Parity Error byte 7	PFSOPI(34)	PFS84/55
40	ALN Micrand Parity Error byte 0	PFSALN(54)	PFS89/8
41	ALN Micrand Parity Error byte 1	PFSALN(55)	PFS89/9
42	ALN Micrand Parity Error byte 2	PFSALN(56)	PFS89/10
43	ALN Micrand Parity Error byte 3	PFSALN(57)	PFS89/11
44	ALN Micrand Parity Error byte 4	PFSALN(58)	PFS89/12
45	ALN Micrand Parity Error byte 5	PFSALN(59)	PFS89/13
46	ALN Micrand Parity Error byte 6	PFSALN(60)	PFS89/14
47	ALN Micrand Parity Error byte 7	PFSALN(61)	PFS89/15
48	Not Used		
49	Not Used		
50	Not Used		
51	Not Used		
52	Processor Did Not Halt Flag from PP Program		
53	Control Store Address bit 0		CSA/53

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
54	Control Store Address bit 1		CSA/54
55	Control Store Address bit 2		CSA/55
56	Control Store Address bit 3		CSA/56
57	Control Store Address bit 4		CSA/57
58	Control Store Address bit 5		CSA/58
59	Control Store Address bit 6		CSA/59
60	Control Store Address bit 7		CSA/60
61	Control Store Address bit 8		CSA/61
62	Control Store Address bit 9		CSA/62
63	Control Store Address bit 10		CSA/63

Section 10, Subsection 0

This subsection tests integer entry to the Divide Remainder register.

Hardware Tested:

- AND gates to transfer Quotient bit 32 or bit 0 to the Division Remainder Least Significant Bit (BIT) signal.
- Minor Cycle Delay registers for Quotient bits 0 and 32.
- Transfer of SNCRO to C register.

Method:

The testing of integer entry into the Divide Remainder register in 32- and 64-bit Integer mode is done by loading the divide arrays in Integer mode, doing three iterations of divide, and then outputting the Quotient register and SNCRO through the C register to the Multiply/Divide Output mux.

The soft control used by this subsection will load the Quotient and Divisor registers, shift the Quotient register, then run the Divide Arrays three times. The first iteration is a divide into all zeros, the second divides into bit 0 or 32, the third divides into bit 1 or 33. The divide arrays are then clocked once more to load the group propagates and group generates of the third iteration into the Divide Carry Interchange. The C register is then loaded from the Divide Quotient register, and the SNCRO signal and the C register are enabled to the Multiply/Divide Output mux.

Microands Used:

Sequence Number	Description
71	Test Quotient transfer through BIT on 64-bit integer.
72	Test Quotient transfer through BIT on 32-bit integer.

Initialization:

- Write Divisor, 0000 0000 0000 0001(HEX), in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
  a. Report test pattern; informational.

C.2 Test Quotient register bit 0 to BIT transfer.
  a. Start processor at Microand Sequence 71; test Quotient transfer through BIT on 64-bit integer.
  b. Wait halt.
  c. Read and report contents of RF0A.
  d. Error if not equal to expected result.

C.3 Read/format and report status.
  a. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Test Quotient register bit 32 to BIT transfer.
  a. Start processor at Microand Sequence 72; test Quotient transfer through BIT on 32-bit integer.
  b. Wait halt.
  c. Read and report contents of RF0A.
  d. Error if not equal to expected result.

C.5 Read/format and report status.
  a. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Report First Failure parity information.
  a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

Iteration Count	Test Pattern	Condition	Expected Result
0	0000 0000 0000 0001	C.2	0000 0000 0000 0140
		C.4	0000 0000 0000 0140
1	4000 0000 0000 0001	C.2	0000 0000 0000 0148
		C.4	0000 0000 0000 0140
2	0000 0000 4000 0001	C.2	0000 0040 0000 0140
		C.4	0000 0040 0000 0148
3	4000 0000 4000 0001	C.2	0000 0040 0000 0148
		C.4	0000 0040 0000 0148

Section 10, Subsection 1

This subsection tests upper Divide arrays.

Hardware Tested:

- Upper five arrays of the Divide logic.

Method:

The testing of the Upper Divide arrays is done by doing a 64-bit integer divide. This involves a load and two's complement of the Quotient register, if negative, and then 67 divide iterations to calculate the result. The Quotient register is then transferred to the C register and output through the Multiply/Divide Output mux.

The ALN soft control that loads and starts the divide is executed one time, the ALN soft control that runs the Divide is executed 16 times, and the ALN soft control that selects the output path is executed one time.

The major cycle control used does the following:

- The first micrand enables loading the Divisor and Quotient as 64-bit integers and starts the divide.
- The second through the seventeenth micrands run the divide and are identical micrands.
- The eighteenth micrand outputs the Multiply/Divide Output mux through the ALN General network and writes the register file.

Micrands Used:

Sequence Number	Description
73	Run 64-bit integer divide.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write divisor from test pattern in RF0C.
a. Report divisor; informational.

C.2 Write quotient for C.3 from test pattern in RF0B.
a. Report quotient; informational.

C.3 Test Integer Divide.
a. Start processor at Micrand Sequence 73; run 64-bit integer divide.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

C.4 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.5 Write quotient for C.6 from test pattern in RF0B.
a. Report quotient; informational.

C.6 Test Integer Divide.
a. Start processor at Micrand Sequence 73; run 64-bit integer divide.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

C.7 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.8 Write quotient for C.9 from test pattern in RF0B.
a. Report quotient; informational.

C.9 Test Integer Divide.
a. Start processor at Micrand Sequence 73; run 64-bit integer divide.
b. Wait halt.
c. Read and report contents of RF0A.
d. Error if not equal to expected result.

C.10 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003(HEX).

C.11 Write quotient for C.12 from test pattern in RF0B.
a. Report quotient; informational.

Conditions:

C.12 Test Integer Divide.
- a. Start processor at Micrand Sequence 73; run 64-bit integer divide.
- b. Wait halt.
- c. Read and report contents of RF0A.
- d. Error if not equal to expected result.

C.13 Read/format and report status.
- a. Error if not equal to 0000 0000 0000 0003(HEX).

C.14 Report First Failure parity information.
- a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal) Listed by Condition:

Iteration Count	Test Pattern Divisor	Condition	Test Pattern Quotient	Expected Result
0	0000 0000 0000 0001	C.3	0000 0000 0000 0000	0000 0000 0000 0000
		C.6	0000 0000 0000 0001	0000 0000 0000 0001
		C.9	0000 0000 0000 0002	0000 0000 0000 0002
		C.12	0000 0000 0000 0004	0000 0000 0000 0004
1	0000 0000 0000 0002	C.3	0000 0000 0000 0001	0000 0000 0000 0000
		C.6	0000 0000 0000 0002	0000 0000 0000 0001
		C.9	0000 0000 0000 0004	0000 0000 0000 0002
		C.12	0000 0000 0000 0008	0000 0000 0000 0004
2	0000 0000 0000 0004	C.3	0000 0000 0000 0002	0000 0000 0000 0000
		C.6	0000 0000 0000 0004	0000 0000 0000 0001
		C.9	0000 0000 0000 0008	0000 0000 0000 0002
		C.12	0000 0000 0000 0010	0000 0000 0000 0004
3	0000 0000 0000 0008	C.3	0000 0000 0000 0004	0000 0000 0000 0000
		C.6	0000 0000 0000 0008	0000 0000 0000 0001
		C.9	0000 0000 0000 0010	0000 0000 0000 0002
		C.12	0000 0000 0000 0020	0000 0000 0000 0004
4	0000 0000 0000 0010	C.3	0000 0000 0000 0008	0000 0000 0000 0000
		C.6	0000 0000 0000 0010	0000 0000 0000 0001
		C.9	0000 0000 0000 0020	0000 0000 0000 0002
		C.12	0000 0000 0000 0040	0000 0000 0000 0004
.	.	.	.	.
.	.	.	.	.
3A	0400 0000 0000 0000	C.3	0200 0000 0000 0000	0000 0000 0000 0000
		C.6	0400 0000 0000 0000	0000 0000 0000 0001
		C.9	0800 0000 0000 0000	0000 0000 0000 0002
		C.12	1000 0000 0000 0000	0000 0000 0000 0004

Iteration Count	Test Pattern Divisor	Condition	Test Pattern Quotient	Expected Result
3B	0800 0000 0000 0000	C.3	0400 0000 0000 0000	0000 0000 0000 0000
		C.6	0800 0000 0000 0000	0000 0000 0000 0001
		C.9	1000 0000 0000 0000	0000 0000 0000 0002
		C.12	2000 0000 0000 0000	0000 0000 0000 0004
3C	1000 0000 0000 0000	C.3	0800 0000 0000 0000	0000 0000 0000 0000
		C.6	1000 0000 0000 0000	0000 0000 0000 0001
		C.9	2000 0000 0000 0000	0000 0000 0000 0002
		C.12	4000 0000 0000 0000	0000 0000 0000 0004
3D	2000 0000 0000 0000	C.3	1000 0000 0000 0000	0000 0000 0000 0000
		C.6	2000 0000 0000 0000	0000 0000 0000 0001
		C.9	4000 0000 0000 0000	0000 0000 0000 0002
		C.12	8000 0000 0000 0000	0000 0000 0000 0004

End of first pattern sequence.

3E	4000 0000 0000 0000	C.3	1000 0000 0000 0000	0000 0000 0000 0000
		C.6	2000 0000 0000 0000	0000 0000 0000 0000
		C.9	4000 0000 0000 0000	0000 0000 0000 0001
		C.12	8000 0000 0000 0000	0000 0000 0000 0002

Start of second pattern sequence:

3F	0000 9000 0000 0000	C.3	0000 9000 0000 0000	0000 0000 0000 0001
		C.6	0001 2000 0000 0000	0000 0000 0000 0002
		C.9	0001 A000 0000 0000	0000 0000 0000 0002
		C.12	0001 B000 0000 0000	0000 0000 0000 0003
40	0001 2000 0000 0000	C.3	0001 2000 0000 0000	0000 0000 0000 0001
		C.6	0002 4000 0000 0000	0000 0000 0000 0002
		C.9	0003 4000 0000 0000	0000 0000 0000 0002
		C.12	0003 6000 0000 0000	0000 0000 0000 0003
41	0002 4000 0000 0000	C.3	0002 4000 0000 0000	0000 0000 0000 0001
		C.6	0004 8000 0000 0000	0000 0000 0000 0002
		C.9	0006 8000 0000 0000	0000 0000 0000 0002
		C.12	0006 C000 0000 0000	0000 0000 0000 0003
42	0004 8000 0000 0000	C.3	0004 8000 0000 0000	0000 0000 0000 0001
		C.6	0009 0000 0000 0000	0000 0000 0000 0002
		C.9	000D 0000 0000 0000	0000 0000 0000 0002
		C.12	000D 8000 0000 0000	0000 0000 0000 0003
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
49	0480 0000 0000 0000	C.3	0480 0000 0000 0000	0000 0000 0000 0001
		C.6	0900 0000 0000 0000	0000 0000 0000 0002
		C.9	0D00 0000 0000 0000	0000 0000 0000 0002
		C.12	0D80 0000 0000 0000	0000 0000 0000 0003

Iteration Count	Test Pattern Divisor	Condition	Test Pattern Quotient	Expected Result
4A	0900 0000 0000 0000	C.3	0900 0000 0000 0000	0000 0000 0000 0001
		C.6	1200 0000 0000 0000	0000 0000 0000 0002
		C.9	1A00 0000 0000 0000	0000 0000 0000 0002
		C.12	1B00 0000 0000 0000	0000 0000 0000 0003
4B	1200 0000 0000 0000	C.3	1200 0000 0000 0000	0000 0000 0000 0001
		C.6	2400 0000 0000 0000	0000 0000 0000 0002
		C.9	3400 0000 0000 0000	0000 0000 0000 0002
		C.12	3600 0000 0000 0000	0000 0000 0000 0003
4C	2400 0000 0000 0000	C.3	2400 0000 0000 0000	0000 0000 0000 0001
		C.6	4800 0000 0000 0000	0000 0000 0000 0002
		C.9	6800 0000 0000 0000	0000 0000 0000 0002
		C.12	6C00 0000 0000 0000	0000 0000 0000 0003
End of second parttern sequence and start of third pattern sequence.				
4D	0000 5000 0000 0000	C.3	0000 9000 0000 0000	0000 0000 0000 0001
		C.6	0000 E000 0000 0000	0000 0000 0000 0002
		C.9	0001 3000 0000 0000	0000 0000 0000 0003
		C.12	0001 9000 0000 0000	0000 0000 0000 0005
4E	0000 A000 0000 0000	C.3	0001 2000 0000 0000	0000 0000 0000 0001
		C.6	0001 C000 0000 0000	0000 0000 0000 0002
		C.9	0002 6000 0000 0000	0000 0000 0000 0003
		C.12	0003 2000 0000 0000	0000 0000 0000 0005
4F	0001 4000 0000 0000	C.3	0002 4000 0000 0000	0000 0000 0000 0001
		C.6	0003 8000 0000 0000	0000 0000 0000 0002
		C.9	0004 C000 0000 0000	0000 0000 0000 0003
		C.12	0006 4000 0000 0000	0000 0000 0000 0005
50	0002 8000 0000 0000	C.3	0004 8000 0000 0000	0000 0000 0000 0001
		C.6	0007 0000 0000 0000	0000 0000 0000 0002
		C.9	0009 8000 0000 0000	0000 0000 0000 0003
		C.12	000C 8000 0000 0000	0000 0000 0000 0005
.	.	.	.	.
.	.	.	.	.
58	0280 0000 0000 0000	C.3	0480 0000 0000 0000	0000 0000 0000 0001
		C.6	0700 0000 0000 0000	0000 0000 0000 0002
		C.9	0980 0000 0000 0000	0000 0000 0000 0003
		C.12	0C80 0000 0000 0000	0000 0000 0000 0005
59	0500 0000 0000 0000	C.3	0900 0000 0000 0000	0000 0000 0000 0001
		C.6	0E00 0000 0000 0000	0000 0000 0000 0002
		C.9	1300 0000 0000 0000	0000 0000 0000 0003
		C.12	1900 0000 0000 0000	0000 0000 0000 0005

Iteration Count	Test Pattern Divisor	Condition	Test Pattern Quotient	Expected Result
5A	0A00 0000 0000 0000	C.3	1200 0000 0000 0000	0000 0000 0000 0001
		C.6	1C00 0000 0000 0000	0000 0000 0000 0002
		C.9	2600 0000 0000 0000	0000 0000 0000 0003
		C.12	3200 0000 0000 0000	0000 0000 0000 0005
5B	1400 0000 0000 0000	C.3	2400 0000 0000 0000	0000 0000 0000 0001
		C.6	3800 0000 0000 0000	0000 0000 0000 0002
		C.9	4C00 0000 0000 0000	0000 0000 0000 0003
		C.12	6400 0000 0000 0000	0000 0000 0000 0005

Section 10, Subsection 2

This subsection tests the Pop Counter.

Hardware Tested:

- Pop Counter.
- V5 and V10 Control Signals.
- Multiply/Divide Output mux for select of Pop Counter.

Method:

The testing of the Pop Counter and its output through the Multiply/Divide Output mux is tested by loading the number into the Divide Quotient register. As the Quotient register is shifted each minor cycle the Pop Counter samples quotient bit 4 and increments each time it is set. The output of the Pop Counter is then enabled through the Multiply/Divide Output mux. The Pop Counter is in bits 58 through 62 of the mux. Therefore, the output is the actual Pop Count times 2.

The Minor Cycle control used loads the Quotient register, runs the Divide arrays to shift the quotient and clock the Pop Counter. After 60 shifts of the Quotient the Pop Count is enabled to the Multiply/Divide Output mux.

The Major Cycle control is as follows:

- Microand 1 enables load of 64-bit quantity into Quotient register.
- Microands 2 through 16 run the Divide arrays and the Pop Counter.
- Microand 18 outputs the Multiply/Divide Output mux through the ALN General network and writes the register file.

Microrands Used:

Sequence Number	Description
61	Test Pop Counter.

Initialization:

- Write 0000 0000 0000 0000(HEX) in RF0C.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Test Pop Counter.
a. Start processor at Microrand Sequence 61; test Pop Counter.
b. Wait halt.
c. Read and report contents of RF0A bits 56 through 63.
d. Error if not equal to expected result.

C.3 Read/format and report status.
a. Error if not equal to 0000 0000 0000 0003.

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns and Expected Results (Hexadecimal):

NOTE

Only the expected result for bits 56 through 63 are shown; bits 0 through 55 are always zeros.

Iteration Count	Test Pattern	Expected Result for Bits 56 through 63
0	AAAA AAAA AAAA AAAA	3C
1	5555 5555 5555 5555	3C
2	0000 0000 0000 0000	00
3	FFFF FFFF FFFF FFFF	78
4	0000 0000 0000 0001	02
5	0000 0000 0000 0003	04
6	0000 0000 0000 0007	06
7	0000 0000 0000 000F	08
8	0000 0000 0000 001F	0A
.	.	.
.	.	.
.	.	.
3C	01FF FFFF FFFF FFFF	72
3D	03FF FFFF FFFF FFFF	74
3E	07FF FFFF FFFF FFFF	76
3F	0FFF FFFF FFFF FFFF	78

SECTION 11 - CHECK MULTIPLY/DIVIDE UNIT FOR ERROR STATUS REPORTING, PARITY CHECKING AND PARITY GENERATION

This section verifies the operation of the B and C register Parity Error detection and reporting logic, the Multiply Carry Error detection and reporting logic, the B and C Register Parity Checking logic, and the Multiply Final Adder Parity Generators.

Subsection 0 - Test C register parity error detection and reporting.

Subsection 1 - Test parity toggle on C register shift data entry.

Subsection 2 - Test C register parity generators and checkers.

Subsection 3 - Test B register parity Error detection and reporting.

Subsection 4 - Test B register parity checkers and Multiply Final Adder parity generators.

Subsection 5 - Test multiply carry error status reporting.

Method:

- See subsection for details.
- Minor Cycle Control:
  - Two types of Minor Cycle Control are used in this subsection to control the Multiply Unit for status testing.
    1. Output C register will load the B and C registers, shift the C register, check parity on the B and C registers, and enable the C register to the Multiply/Divide Output mux.
    2. Output of the MSUM bits will load the B and C registers, strobe parity on the B and C registers, shift the C register, discard the results of the first multiply, clock the results of the second multiply into the Multiply Partial Summing network, and enable MSUM bits 8 through 71 to the Multiply/Divide Output mux. After MSUM bits 8 through 71 have been output through the ALN General network, MSUM bits 0 through 7 are enabled to the Multiply/Divide Output mux and output through the ALN General network.

Status Reporting:

All status listed in the Section Status Format will be read/formatted and checked by all subsections in this section.

Initialization:

Initialization is performed as part of subsection 0 PP code.

- Write the following ALN Soft Control Memory Map (hexadecimal) into the locations specified.

Location	32-bit Value to be Written
0	5188 4188
1	C108 D308
2	0108 0108
3	0000 0000
4	0000 0114
5	0000 0000
6	0000 0000
7	0000 0000
8	0000 010C
9	0000 0000
A	0000 0000
B	0000 0000

Section Status Format:

The following bit assignments are used by all conditions that report status.

NOTE

Any status not read by a condition and all unused bit positions will be reported as zero. Refer to Section Method for status that will be read and reported.

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
0	Soft Control Register Parity Error byte 0	PFSALN(62)	PFS88/14
1	Soft Control Register Parity Error byte 1	PFSALN(63)	PFS88/15
2	Not Used		
3	Not Used		
4	Not Used		
5	Not Used		
6	Multiply Carry Miscompare byte 0	PFSALN(44)	PFS88/60

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
7	Multiply Carry Miscompare byte 1	PFSALN(45)	PFS88/61
8	Not Used		
9	Multiply Carry Miscompare byte 2	PFSALN(46)	PFS89/0
10	Multiply Carry Miscompare byte 3	PFSALN(47)	PFS89/1
11	Multiply Carry Miscompare byte 4	PFSALN(48)	PFS89/2
12	Multiply Carry Miscompare byte 5	PFSALN(49)	PFS89/3
13	Multiply Carry Miscompare byte 6	PFSALN(50)	PFS89/4
14	Multiply Carry Miscompare byte 7	PFSALN(51)	PFS89/5
15	Multiply Carry Miscompare byte 8	PFSALN(52)	PFS89/6
16	Multiply C Register Parity Error byte 0	PFSALN(0)	PFS88/16
17	Multiply C Register Parity Error byte 1	PFSALN(1)	PFS88/17
18	Multiply C Register Parity Error byte 2	PFSALN(2)	PFS88/18
19	Multiply C Register Parity Error byte 3	PFSALN(3)	PFS88/19
20	Multiply C Register Parity Error byte 4	PFSALN(4)	PFS88/20
21	Multiply C Register Parity Error byte 5	PFSALN(5)	PFS88/21
22	Multiply C Register Parity Error byte 6	PFSALN(6)	PFS88/22
23	Multiply C Register Parity Error byte 7	PFSALN(7)	PFS88/23
24	Multiply B Register Parity Error byte 0	PFSALN(8)	PFS88/24
25	Multiply B Register Parity Error byte 1	PFSALN(9)	PFS88/25
26	Multiply B Register Parity Error byte 2	PFSALN(10)	PFS88/26
27	Multiply B Register Parity Error byte 3	PFSALN(11)	PFS88/27
28	Multiply B Register Parity Error byte 4	PFSALN(12)	PFS88/28
29	Multiply B Register Parity Error byte 5	PFSALN(13)	PFS88/29
30	Multiply B Register Parity Error byte 6	PFSALN(14)	PFS88/30
31	Multiply B Register Parity Error byte 7	PFSALN(15)	PFS88/31
32	DAI Data Parity Error byte 0	PFSOPI(27)	PFS84/48
33	DAI Data Parity Error byte 1	PFSOPI(28)	PFS84/49

Conditions Report Bit Number	Status Reported	Signal Name	Register/Bit
34	DAI Data Parity Error byte 2	PFSOPI(29)	PFS84/50
35	DAI Data Parity Error byte 3	PFSOPI(30)	PFS84/51
36	DAI Data Parity Error byte 4	PFSOPI(31)	PFS84/52
37	DAI Data Parity Error byte 5	PFSOPI(32)	PFS84/53
38	DAI Data Parity Error byte 6	PFSOPI(33)	PFS84/54
39	DAI Data Parity Error byte 7	PFSOPI(34)	PFS84/55
40	ALN Microrand Parity Error byte 0	PFSALN(54)	PFS89/8
41	ALN Microrand Parity Error byte 1	PFSALN(55)	PFS89/9
42	ALN Microrand Parity Error byte 2	PFSALN(56)	PFS89/10
43	ALN Microrand Parity Error byte 3	PFSALN(57)	PFS89/11
44	ALN Microrand Parity Error byte 4	PFSALN(58)	PFS89/12
45	ALN Microrand Parity Error byte 5	PFSALN(59)	PFS89/13
46	ALN Microrand Parity Error byte 6	PFSALN(60)	PFS89/14
47	ALN Microrand Parity Error byte 7	PFSALN(61)	PFS89/15
48	Not Used		
49	Not Used		
50	Not Used		
51	Not Used		
52	Processor Did Not Halt Flag from PP Program		
53	Control Store Address bit 0		CSA/53
54	Control Store Address bit 1		CSA/54
55	Control Store Address bit 2		CSA/55
56	Control Store Address bit 3		CSA/56
57	Control Store Address bit 4		CSA/57
58	Control Store Address bit 5		CSA/58
59	Control Store Address bit 6		CSA/59
60	Control Store Address bit 7		CSA/60
61	Control Store Address bit 8		CSA/61
62	Control Store Address bit 9		CSA/62
63	Control Store Address bit 10		CSA/63

Section 11, Subsection 0

This subsection tests C register Parity Error detection and reporting.

Hardware Tested:

- C register parity logic.
- ALN Microrand aux for C register parity error status reporting.

Method:

The testing of the C register parity error detection and reporting logic is performed by writing the test pattern into the register File with bad parity, transferring it to the C register, checking parity in the C register, shifting the C register 12 places, and outputting the C register through the Multiply/Divide Output mux. This value is copied through the ALN General network so that parity can also be checked at DAI. Since the C register regenerates parity there should not be a DAI parity error.

Microrands Used:

Sequence Number	Description
62	Test C register parity using unsigned entry.

Initialization:

- Write 0000 0000 0000 0000(HEX) in RF0B.
- Start processor at Microrand Sequence 62 before each pattern in pattern loop to clear previous error.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0C with bad parity.
a. Report test pattern; informational.

C.2 Test C register parity errors.
a. Start processor at Microrand Sequence 62; test C register parity using unsigned entry.
b. Wait halt.
c. Read/format and report status.
d. Error if not equal to expected result.

C.3 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Post-Subsection operations:

- Write 0000 0000 0000 0000(HEX) in RF0C to remove any bad parity.
- Start processor at Microrand Sequence 62 to clear any errors in Multiply/Divide unit.

Test Patterns and Expected Results (Hexadecimal):

Iteration Count	Test Pattern	Expected Result
0	FE00 0000 0000 0000	0000 8000 0000 0003
1	00FE 0000 0000 0000	0000 4000 0000 0003
2	0000 FE00 0000 0000	0000 2000 0000 0003
3	0000 00FE 0000 0000	0000 1000 0000 0003
4	0000 0000 FE00 0000	0000 0800 0000 0003
5	0000 0000 00FE 0000	0000 0400 0000 0003
6	0000 0000 0000 FE00	0000 0200 0000 0003
7	0000 0000 0000 00FE	0000 0100 0000 0003

Section 11, Subsection 1

This subsection tests parity toggle on C register shift data entry.

Hardware Tested:

- C register parity generators.
- PTM bit 16.

Method:

The testing of the parity toggle on the C register shift data entry using PTM bit 16 is performed by loading the C register with good parity, then shifting it one time and toggling the parity. The C register is then output through the Multiply/Divide Output mux and the ALN General network. Status will be read and checked and parity errors should be seen at DAI.

Microrands Used:

Sequence Number	Description
62	Test C register parity using unsigned entry.

Initialization:

- Clear PTM bit 16.
- Start processor at Microrand Sequence 62 before each pattern in pattern loop to clear previous errors.
- Set PTM bit 16.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0C.
a. Report test pattern; informational.

C.2 Test C register parity toggle.
a. Start processor at Microrand Sequence 62; test C register parity using unsigned entry.
b. Wait halt.
c. Read/format and report status.
d. Error if not equal to 0000 0000 FF00 0003(HEX).

C.3 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Post-Subsection operations:

- Write 0000 0000 0000 0000(HEX) in RF0A to remove any parity errors.
- Clear PTM bit 16.
- Start processor at Microrand Sequence 62 to clear any errors in the Multiply/Divide unit.

Test Patterns (Hexadecimal):

Iteration Count	Test Pattern
0	FE00 0000 0000 0000
1	00FE 0000 0000 0000
2	0000 FE00 0000 0000
3	0000 00FE 0000 0000

Iteration Count	Test Pattern
4	0000 0000 FE00 0000
5	0000 0000 00FE 0000
6	0000 0000 0000 FE00
7	0000 0000 0000 00FE

Section 11, Subsection 2

This subsection tests C register parity generators and checkers.

Hardware Tested:

- C register parity generators and checking logic.

Method:

The testing of the C register parity generators and checkers is performed by using an ascending test pattern of 0 through 255 in each byte to ensure that a false parity error is not reported.

The test pattern is loaded into the C register, parity is checked on it, and it is then shifted 12 places, enabled to the Multiply/Divide Output mux, and copied through the ALN General network.

Microrands Used:

Sequence Number	Description
62	Test C register parity using unsigned entry.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0C.
a. Report test pattern; informational.

Conditions:

C.2 Test C register parity generators and checkers.
- a. Start processor at Micrand Sequence 62; test C register parity using unsigned entry.
- b. Wait halt.
- c. Read/format and report status.
- d. Error if not equal to 0000 0000 0000 0003(HEX).

C.3 Report First Failure parity information.
- a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns (Hexadecimal):

Iteration Count	Test Pattern
0	0000 0000 0000 0000
1	0101 0101 0101 0101
2	0202 0202 0202 0202
3	0303 0303 0303 0303
.	.
.	.
.	.
FC	FCFC FCFC FCFC FCFC
FD	FDFD FDFD FDFD FDFD
FE	FEFE FEFE FEFE FEFE
FF	FFFF FFFF FFFF FFFF

Section 11, Subsection 3

This subsection tests B register parity error detection and reporting.

Hardware Tested:

- B register parity error logic.
- ALN micrand aux for parity error status reporting.

Method:

The testing of the B register parity error detection and reporting is performed by writing the test pattern into the register file with bad parity, and transferring the register file to the B register where the bad parity should be detected and reported.

Micrands Used:

Sequence Number	Description
63	Test B register and Multiply Final Adder parity using unsigned entry.

Initialization:

- Write 0000 0000 0000 0800(HEX) in RF0C.
- Start processor at Micrand Sequence 63 before each pattern in pattern loop to remove previous error.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern with bad parity in RF0B.
- a. Report test pattern; informational.

C.2 Test B register parity error detection.
- a. Start processor at Micrand Sequence 63; test B register and Multiply Final Adder parity using unsigned entry.
- b. Wait halt.
- c. Read/format and report status.
- d. Error if not equal to expected result.

C.3 Report First Failure parity information.
- a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Post-Subsection operations:

- Write 0000 0000 0000 0000(HEX) in RF0B to remove any parity error.
- Start processor at Microrand Sequence 63 to clear errors in Multiply/ Divide unit.

Test Patterns and Expected Results (Hexadecimal):

Iteration Count	Test Pattern	Expected Result
0	FE00 0000 0000 0000	0000 0080 0000 0003
1	00FE 0000 0000 0000	0000 0040 0000 0003
2	0000 FE00 0000 0000	0000 0020 0000 0003
3	0000 00FE 0000 0000	0000 0010 0000 0003
4	0000 0000 FE00 0000	0000 0008 0000 0003
5	0000 0000 00FE 0000	0000 0004 0000 0003
6	0000 0000 0000 FE00	0000 0002 0000 0003
7	0000 0000 0000 00FE	0000 0001 0000 0003

Section 11, Subsection 4

This subsection tests B register parity checkers and Multiply Final Adder parity generators.

Hardware Tested:

- B register parity checkers.
- Multiply Final Adder parity generators.

Method:

The testing of the B register parity checkers and the Multiply Final Adder parity generators is performed by using an ascending pattern of 0 through 255 in each byte of the test pattern. The test pattern is loaded into the B register and parity is checked. Then it is multiplied by one of two multipliers. One multiplier routes the test pattern to MSUM bits 8 through 71 and the other routes the test pattern to MSUM bits 0 through 63. MSUM bits 0 through 71 are output through the Multiply/Divide Output mux and copied through the ALN General network so that the generated parity from the Multiply Final Adder can be checked at DAI.

Microrands Used:

Sequence Number	Description
63	Test B register and Multiply Final Adder Parity using unsigned entry.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Write multiplier, 0000 0000 0000 0800(HEX), in RF0C.
a. Report multiplier; informational.

C.3 Test B register parity checkers and MSUM bits 8 through 71 parity generation.
a. Start processor at Microrand Sequence 63; test B register parity and Multiply Final Adder parity using unsigned entry.
b. Wait halt.
c. Read/format and report status.
d. Error if not equal to 0000 0000 0000 0003(HEX).

C.4 Write multiplier, 0000 0000 0008 0000(HEX), in RF0C.
a. Report multiplier; informational.

C.5 Test B register parity checkers and MSUM bits 0 through 63 parity generation.
a. Start processor at Microrand Sequence 63; test B register and Multiply Final Adder parity using unsigned entry.
b. Wait halt.
c. Read/format and report status.
d. Error if not equal to 0000 0000 0000 0003(HEX).

C.6 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

If RI is set, go to C.2.

Exit subsection if no more test patterns.

Test Patterns (Hexadecimal):

Iteration Count	Test Pattern
0	0000 0000 0000 0000
1	0101 0101 0101 0101
2	0202 0202 0202 0202
3	0303 0303 0303 0303
.	.
.	.
FC	FCFC FCFC FCFC FCFC
FD	FDFD FDFD FDFD FDFD
FE	FEFE FEFE FEFE FEFE
FF	FFFF FFFF FFFF FFFF

Section 11, Subsection 5

This subsection tests multiply carry miscompare status reporting.

Hardware Tested:

- Multiply Final Adder carry error logic.
- ALN Micrand aux for multiply carry miscompare status reporting.
- V11 enable of carry error test.

Method:

The testing of Multiply Final Adder carry miscompare logic is performed by using PTM bit 16 to force a carry miscompare. Test pattern 0 will check for all carry miscompares 0 through 8 and test patterns 1 through 8 will test carry miscompares 1 through 8, one at a time. This will also cause DAI parity errors because of the C register parity toggle. The output of MSUM bits 0 through 7 causes C register bits 0 through 47 to be output with MSUM bits 0 through 7.

Micrands Used:

Sequence Number	Description
63	Test B register and Multiply Final Adder parity using unsigned entry.

Initialization:

- Clear PTM bit 16.
- Write multiplier, 0000 0000 0000 0000(HEX), in RF0C.
- Start processor at Micrand Sequence 63 before each pattern in pattern loop to clear previous error.
- Set PTM bit 16.

Conditions:

The following conditions are repeated for each test pattern.

C.0 Report iteration count; informational.

C.1 Write test pattern in RF0B.
a. Report test pattern; informational.

C.2 Change multiplier at pattern 1.
a. If pattern 1, write new multiplier, 0000 0000 0001 FC00(HEX), in RF0C.
b. Report multiplier; informational.

C.3 Test multiply carry miscompares.
a. Start processor at Micrand Sequence 63; test B register parity and multiply Final Adder parity using unsigned entry.
b. Wait halt.
c. Read/format and report status.
d. Error if not equal to expected result.

C.4 Report First Failure parity information.
a. Report PFS87 bits 19 - 23 in bits 59 - 63; information only.

Exit subsection if no more test patterns.

Post-Subsection Operations:

- Clear PTM bit 16.
- Start processor at Micrand Sequence 63 to clear any error in Multiply/Divide unit.

Test Patterns and Expected Results (Hexadecimal):

Iteration Count	Test Pattern	Expected Result
0	0000 0000 0000 0000	037F 0000 FC00 0003
1	0F00 0000 0000 0000	0100 0000 FC00 0003
2	000F 0000 0000 0000	0040 0000 FC00 0003
3	0000 0F00 0000 0000	0020 0000 FC00 0003
4	0000 000F 0000 0000	0010 0000 FC00 0003
5	0000 0000 0F00 0000	0008 0000 FC00 0003
6	0000 0000 000F 0000	0004 0000 FC00 0003
7	0000 0000 0000 0F00	0002 0000 FC00 0003
8	0000 0000 0000 000F	0001 0000 FC00 0003

THIS PAGE HAS INTENTIONALLY BEEN LEFT BLANK

Abs

Absolute.

AC or A/C

Address Control.

ACMIC

Address Control Functional Micrand.

ADATB/RDSB/B

Refers to B Operand data in Multiply/ Divide unit test MDT3/P.

ADATC/RDSC/C

Refers to C Operand data in Multiply/ Divide unit test MDT3/P.

Adder

Refers to Large Adder in ALN test ANT3/P. Refer to Multiply Final Adder in test MDT3/P.

Adder Bit 47 FF

Large Adder Result bit 47 flip-flop in ALN test ANT3/P.

Adrs

Address.

ALN

Arithmetic and Logical Network.

ALU

Arithmetic and Logical Unit.

AOR

Address-Out-of-Range.

Arith

Arithmetic.

ASCII

American Standard Code for Informa- tion Interchange.

ASE

Address Specification Error.

ASID

Active Segment Identifier.

Assy

Assembly.

Aux

Auxiliary.

B

Refers to the operand input to ALN on the RSDB data path. Used in ALN test ANT3/P.

BCD

Binary Coded Decimal.

BD

Branch Delta.

BDP

Business Data Processor.

Bfr

Buffer.

Bin

Binary.

Bkpt

Breakpoint.

1BN

Byte Number.

Bound

Boundary.

Br

Branch.

C

Refers to the operand input to the ALN on the RDSC data path. Used in ALN test ANT3/P.

CA

Condition Action.

CABR

Cache Address Buffer Register.

CAE

Condition Action Extension.

CAR

Cache Address Register.

CBP

Code Base Pointer.

CCR

Clock Condition Register.

CEL

Corrected Error Log.

C.GT.B

C Greater than B signal from the SKG in ALN test ANT3/P.

Chan

Channel.

Clk

Clock.

Clr

Clear.

CM

Central Memory.

CMC

Central Memory Control.

Cnt

Count.

Cntr

Counter.

Coef

Coefficient.

Com

Common.

Comp

Complement.

Cond

Condition.

Confl

Conflict.

Cont

Control.

Conv

Convert, Converter.

Cor

Corrected.

CPU

Central Processing Unit.

1CS

Condition Sensing.

CSA

Control Store Address.

CST

Control Store.

CW1

Control Word 1.

CW2

Control Word 2.

DAI

Data Interchange.

DCDR

Decoder.

DCI

Data control information. Two bytes of information that give the maintenance register address.

DEC

Dependent Environment Control, Decimal.

Decr

Decrement.

Descr

Descriptor.

Destn

Destination.

Disassy

Disassembly.

Div

Divide.

Divisor Rgtr

Divide network register internal to LSI CI divide arrays.

Dlyd

Delayed.

DMR

Debug Mask Register.

Dsbl

Disable.

DSS

Data Subfunction Select Microand Field.

EBCDIC

Extended Binary Coded Decimal Inter- change Code.

ECC

Error Correction Code.

ECR

C170 Exit Mode Condition Register.

EI

Environment Interface.

EMMR or EMR

Exit Mode Mask Register.

Enbl

Enable.

Encdr

Encoder.

ESE

Environment Specification Error.

EXCH

Exchange.

Exp

Exponent.

Exponent Aux

Exponent Auxiliary board in ALN.

Ext

External.

Fctn

Function.

FF

Flip-flop.

FIFO

First In/First Out.

FLC

C170 Mode Central Memory Field Length.

FNO

Fanout.

FRC

Free Running Counter.

FU

Functional Unit.

Gen

Generator.

Genl

General.

GME

Used by test CTT3/P to indicate a field (bits 62 and 63) within the general micrand.

Hldg

Holding.

i

Operand Register Designator.

IACY

C170 Parcel Select Multiplexer (IFT3/P test).

IAHY

Instruction Assembly Hybrid Multi- plexer (IFT3/P test).

IAPL

C180 Parcel Select Multiplexer (IFT3/P test).

IB02

Instruction Buffer Rank 02 (IFT3/P test).

IB12 valid

Used in ALN test ANT3/P. Instruction Buffer Rank 12 valid signal from Instruction Fetch. This goes active when IF has finished fetching the next instruction from CM.

ICC

Instruction Completion Control.

ICP

Instruction Control Pipeline.

IDS

Immediate Data Select Micrand Field.

IDX

Indexed.

IF

Instruction Fetch.

II

Instruction Issue.

Immed

Immediate.

Incr

Increment.

Indef

Indefinite.

Inf

Infinite.

Inh

Inhibit.

Inp

Input.

Instr

Instruction.

Int

Internal.

Integer Mux

Integer/Exponent Multiplexer in ALN.

Intrpt

Interrupt.

IOU

Input/Output Unit.

ISE

Instruction Specification Error.

IU

Instruction Unit.

j

Operand Register Designator.

JPS

Job Process State.

k

Operand Register Designator.

K

170 18-Bit Immediate Operand.

K/L

Key/Lock.

Kypt

Keypoint.

l

BDP Operand Length.

Ld

Load.

LM

Local Memory.

LMMIC

Local Memory Functional Microrand field.

LOS

Loss of Significance.

LRR

Used in ALN test ANT3/P and Multiply/ Divide test MDT3/P, and ICT3/P to indicate a live register read.

LRU

Least Recently Used.

LRW

Used in ALN test ANT3/P and Multiply/ Divide test MDT3/P and ICT3/P to indicate a live register write.

SD

Least Significant Digit.

LSI

Large Scale Integration.

LSM

Last Symbol Mask.

LSMPRE

LOAD/STORE Multiple Address Adder.

Lwr

Lower.

MAC

Maintenance Access Control.

Maint

Maintenance.

MAR

Microand Address Register.

MBE

Multiple Bit Error.

MC

Master Clear.

MCH

Maintenance Channel.

MCMR

Monitor Condition Mask Register. Also refer to MMR.

MCR

Monitor Condition Register.

Mem

Memory.

Micr

Microand.

Microand Constant

Contents of ALN FU Microand bits 0 through 6.

MIS

Miscellaneous Microand Field.

Misc

Miscellaneous.

MMR

Monitor Mask Register.

MOP

Micro operation.

MPS

Monitor Process State.

MR

Maintenance Register.

MSB

Most Significant Bit.

MSC

Microand Sequence Control. A field (bits 0 through 15) within the general microand.

MSD

Most Significant Digit.

MSNZB

Most Significant Nonzero Byte.

Mult

Multiply.

Mux

Multiplexer.

Neg

Negative.

Norm

Normalize.

NPGK

New P Global Key.

NPLK

New P Local Key.

ns

Nanosecond.

o

BDP Operand Address.

Op

Operand.

Opcode

Operation Code.

OPGK

Old P Global Key.

OPLK

Old P Local Key.

OPI

Operand Issue.

Opn

Operation.

OSB

Operating System Bounds.

Out

Output.

Ovfl

Overflow.

P

Program Address.

Par

Parity.

PDM

Processor Detected Malfunction.

PE

Parity Error.

PFA

Page Frame Address.

PFS

Processor Fault Status.

Ph

Phase.

PIT

Process Interval Timer.

PMF

Performance Monitoring Facility.

PN

Page Number.

PO

Page Offset.

PONR

Point Of No Return.

Pos

Positive.

PP

Peripheral Processor.

Prev

Previous.

Pri

Priority.

Prod

Product.

PSM

Page Size Mask.

PSWF

Page Table Search Without Find.

PTA

Page Table Address.

PTD

Page Table Descriptor.

PTL

Page Table Length.

PTM

Processor Test Mode.

PVA

Process Virtual Address.

PW

Partial Write.

Q

C180 16-Bit Immediate Operand.

Quad

Quadrant.

Quotient Rgtr

Divide network register internal to the LSI CI arrays.

R/W

Read/Write.

RAC

C170 Mode Central Memory Reference Address.

RAM

Random Access Memory.

RCL

Read and Clear Lock.

Rcvr

Receiver.

RDS

Register/Data Select.

RDSA

In ALN test ANT3/P, refers to the micrand field that provides the register file address on a write of ALN data and to the data path through DAI for that data.

RDSB

In ALN test ANT3/P, refers to the following data path to ALN. The specified register file is output to the RDSB bus, transferred to the ALN on ADATB, and is passed through the multiple/divide unit and output multiplexer to the general network.

RDSC

In ALN test ANT3/P, refers to following data path. The specified register file is sent to the RDSC bus, trans- ferred to ADATC, then goes to the general network.

Ref

Reference.

Regen

Regenerator.

Rel

Relative.

Rem

Remainder.

Req

Request.

Resync

Resynchronize, Resynchronizer, Resynchronization.

RF

Register File.

RFOx

Used in ALN test ANT3/P and Multiply/ Divide test MDT3/P to identify the register file being used. x is 5, 6, 7, 8, 9, A, B, C, D, E or F (Hexadecimal).

Rgtr

Register.

RMA

Real Memory Address.

ROM

Read Only Memory.

RSL

Read and Set Lock.

RS64

Right Shift 64 signal from the SKG.

Rxx

Identifies ranks in IF and ICP in Test CTT3/P.

SBE

Single Bit Error.

SC

Soft Control.

SCM

Soft Control Memory.

SCT

Special Character Table.

SECDED

Single Error Correction/Double Error Detection.

Seg

Segment.

Sel

Select.

Sep

Separate.

Seq

Sequence.

Shift Mux

Shift Network Multiplexer.

Shifter

Shift Network.

Sig

Significant.

SIT

System Interval Timer.

SKG

Shift Count Generator.

SM

Segment Map.

SMMIC

Segment Map Functional Microrand Field.

SMO(x)

Segment Descriptor Multiplexer Output bit x.

SMPDM

Segment Map Processor Detected Malfunction.

SN

Segment Number.

Spec

Specification.

SPID

Segment/Page Identifier.

STA

Segment Table Address.

STL

Segment Table Length.

Str

Stream.

Subtr

Subtract.

Suppr

Suppression.

SV

Specification Value.

SVA

System Virtual Address.

Sw

Switch.

Syn

Syndrome.

Sync

Synchronize.

Termn

Terminate.

TFA

Tag File Address.

ubit

Microrand Bit.

UCEL1

Uncorrected Error Log 1.

UCEL2

Uncorrected Error Log 2.

UCR

User Condition Register.

UMR

User Mask Register.

Unbr

Unbranch.

Uncond

Unconditional.

Uncor

Uncorrectable.

Undfl

Underflow.

Unlog

Unlogged.

Upr

Upper.

usec

Microsecond.

UTP

Untranslatable Pointer.

Ux

Used in Multiply/Divide test MDT3/P to identify Major Cycle Control bits 0 through 14.

VMID

Virtual Machine Identifier.

Vx

Used in Multiply/Divide test MDT3/P to identify Minor Cycle Control bits 0 through 15.

DS

Write Data Select.

WOI

Word of Interest.

Xfer

Transfer.

Xltr

Translator.

Xmtr

Transmitter.

XOR

Exclusive OR.

ZIF

Zero Insertion Force.

ZF

Zero Flag.

The following common micrand controls are used within each micrand sequence unless otherwise specified by the micrand sequence.

A. Start or run Multiply/Divide Unit micrand.
   - Transfer RF0B to ADATB.
   - Transfer RF0C to ADATC.
   - Issue ALN Functional Unit micrand that includes the following control:
     - Mux Code = E(HEX).
     - ALN Response.
   - Write RF05.

B. Copy Multiply/Divide Output mux through ALN General network micrand.
   - Transfer RF0B to ADATB.
   - Transfer RF0C to ADATC.
   - Issue following ALN Functional Unit micrand that includes the following controls:
     - Mux Code = 4.
     - SKG Code = 6.
     - ALN Response.
   - Write RF0A.

Micrand Sequence Number	Short Description of Sequence followed by Individual Micrand Description as indicated by X, X+1, and so forth
1	Load ALN idle state and ALN soft control entry point zero into Minor Cycle Control register.
	X — Issue to address control.
	X+1 — Start or run Multiply/Divide common with entry point equal to 0.
2	Load Minor Cycle Control register from ALN soft control entry point equal to 0.
	X — Start or run Multiply/Divide common with entry point equal to 0.
3	Load Minor Cycle Control register from ALN soft control entry point 10(HEX).
	X — Start or run Multiply/Divide common with entry point equal to 10(HEX).
4	Load Minor Cycle Control register from ALN soft control entry point xx.
	X — Start or run Multiply/Divide common with entry point equal to xx. (The soft control entry point is written into the micrand by the PP program.)
5	Load C register with 64 bits and sign extend bit 0.
	X — Start or run Multiply/Divide common with entry point equal to 0; U4, U5, U6; and Multiply/Divide start.
	X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1.
6	Load C register with 64 bits and sign extended bit 32.
	X — Start or run Multiply/Divide common with entry point equal to 0; U6 and Multiply/Divide start.
	X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1.
7	Load C register with input bits 0 through 15 and 32 through 63 and sign extended from bit 0 or 32.
	X — Start or run Multiply/Divide common with entry point equal to 10(HEX); U4, U6; and Multiply/Divide start.
	X+1 — Copy Multiply/Divide Output mux with entry point equal to 1.

Micrand Sequence Number — Short Description of Sequence followed by Individual Micrand Description as indicated by X, X+1, and so forth

8 Load C register with 48 bits and zero fill.

- X — Start or run Multiply/Divide common with entry point equal to 10(HEX); U5, U6; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1.

9 Load C register with 48 bits, sign extended bit 0 or 1, 170 mode.

- X — Start or run Multiply/Divide common with entry point equal to 10(HEX); U5, U6, U8; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1.

10 Run Multiply from ALN soft control entry points 0, 1 and 2.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0 through U6, and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1.
- X+2 — Copy Multiply/Divide Output common, except write RF09 with entry point equal to 2.

11 Run Multiply from ALN soft control entry points 10(HEX), 11(HEX), and 12(HEX).

- X — Start or run Multiply/Divide common with entry point equal to 10(HEX); U0 through U6, and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 11(HEX).
- X+2 — Copy Multiply/Divide Output mux common, except write RF09 with entry point equal to 12(HEX).

12 Load C register signed/unsigned.

- X — Start or run Multiply/Divide common with entry point equal to 11(HEX); U4, U5, U6, U13 and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1.

13 Load C register with 64 bits unsigned.

- X — Start or run Multiply/Divide common with entry point equal to 11(HEX); U5, U6, U13 and Multiply/Divide start.
- X+1 — Copy Multiply/Divide common with entry point equal to 1.

Micrand Sequence Number — Short Description of Sequence followed by Individual Micrand Description as indicated by X, X+1, and so forth

14 Run 32-bit B register multiply from ALN soft control entry points 0, 1 and 2.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0, U3 through U6; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide common with entry point equal to 1.
- X+2 — Copy Multiply/Divide Output mux, except write RF09 with entry point equal to 2.

15 Run 48-bit B register multiply from ALN soft control entry points 0, 1 and 2.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0, U2 through U6; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1.
- X+2 — Copy Multiply/Divide Output mux common, except write RF09 with entry point equal to 2.

16 Run signed/unsigned B register multiply from ALN soft control entry points 10(HEX), 11(HEX) and 12(HEX).

- X — Start or run Multiply/Divide common with entry point equal to 10(HEX); U0 through U3, U5, U6, U13; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide common with entry point equal to 11(HEX).
- X+2 — Copy Multiply/Divide Output mux common, except write RF09 with entry point equal to 12(HEX).

17 Run unsigned multiply from ALN soft control entry points 10(HEX), 11(HEX), and 12(HEX).

- X — Start or run Multiply/Divide common with entry point equal to 10(HEX); U0, U2, U3, U5, U6, U13; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 11(HEX).
- X+2 — Copy Multiply/Divide Output mux common, except write RF09 with entry point equal to 12(HEX).

Micrand Sequence Number	Short Description of Sequence followed by Individual Micrand Description as indicated by X, X+1, and so forth
18	Run loop MSUM bits 8 through 71 to B register from ALN soft control entry points 10(HEX), 11(HEX), and 12(HEX).
	X Start or run Multiply/Divide common with entry point equal to 10(HEX); U0, U2, U3, U5, U6, U13; and Multiply/Divide start.
	X+1 Start or run Multiply/Divide common, except RF07 to ADATB and RF08 to ADATC, with entry point equal to 10(HEX); U5, U6, and Multiply/Divide start.
	X+2 Copy Multiply/Divide Output mux common with entry point equal to 11(HEX).
	X+3 Copy Multiply/Divide Output mux common, except write RF09 with entry point equal to 12(HEX).
19	Run 170 multiply from ALN soft control entry points 0, 1, and 2.
	X Start or run Multiply/Divide common with entry point equal to 0; U0 through U3, U5, U6, U8; and Multiply/Divide start.
	X+1 Copy Multiply/Divide Output mux common with entry point equal to 1.
	X+2 Copy Multiply/Divide Output mux common, except write RF09 with entry point equal to 2.
20	Run 170 B register multiply from ALN soft control entry points 0, 1, and 2.
	X Start or run Multiply/Divide common with entry point equal to 0; U0, U2, U3, U5, U6, U8; and Multiply/Divide start.
	X+1 Copy Multiply/Divide Output mux common with entry point equal to 1.
	X+2 Copy Multiply/Divide Output mux common, except write RF09 with entry point equal to 2.
21	Run 170 multiply for complement correction check from ALN soft control entry points 10(HEX), 1, and 2.
	X Start or run Multiply/Divide common with entry point equal to 10(HEX); U0, U2, U3, U5, U6, U8 and Multiply/Divide start.
	X+1 Copy Multiply/Divide Output mux common with entry point equal to 1.
	X+2 Copy Multiply/Divide Output mux common, except write RF09 with entry point equal to 2.
22	Run multiply and output the C register.
	X Start or run Multiply/Divide common with entry point equal to 0; U0 through U6 and Multiply/Divide start.
	X+1 Start or run Multiply/Divide common with entry point equal to 1; U0 through U6 and Multiply/Divide start.
	X+2 Copy Multiply/Divide Output mux common with entry point equal to 11(HEX).

Sequence Number	Short Description of Sequence followed by Individual Micrand Description as indicated by X, X+1, and so forth
23	Clear UCR.
	X Live register read and clear of the UCR.
24	Run 64-bit integer multiply overflow and read/clear UCR.
	X Start or run Multiply/Divide common with entry point equal to 0; U0 through U6, U12 and Multiply/Divide start.
	X+1 Start or run Multiply/Divide common with entry point equal to 1; U0 through U6, U12 and Multiply/Divide start.
	X+2 Copy Multiply/Divide Output mux common with entry point equal to 11(HEX); Error Code = 2 (multiply overflow); Multiply/Divide run; and Clock Condition register.
	X+3 No-op micrand.
	X+4 No-op micrand.
	X+5 No-op micrand.
	X+6 Live register read and clear of the UCR to RF0D.
25	Run 32-bit integer multiply overflow and read/clear UCR.
	X Start or run Multiply/Divide common with entry point equal to 0; U0, U3, U6 and Multiply/Divide start.
	X+1 Start or run Multiply/Divide common with entry point equal to 2; U0, U3, U6 and Multiply/Divide start.
	X+2 Start or run Multiply/Divide common with entry point equal to 1; U0, U3, U6 and Multiply/Divide start.
	X+3 Copy Multiply/Divide Output mux common with entry point equal to 11(HEX); Error Code = 2 (multiply overflow), Multiply/Divide run and Clock Condition register.
	X+4 No-op micrand.
	X+5 No-op micrand.
	X+6 No-op micrand.
	X+7 Live register read and clear of the UCR to RF0D.
26	Run 180 multiply and encode MFOVF.
	X Start or run Multiply/Divide common with entry point equal to 0; U0 through U6 and Multiply/Divide start.
	X+1 Copy Multiply/Divide Output mux common, except Mux Code = 7 with entry point equal to 1; Normalize Code = 3 (encode MFOVF), Shift Count Port = 1 (Shift according to Significance Count); and Multiply/Divide run.

Microrand Sequence Number	Short Description of Sequence followed by Individual Microrand Description as indicated by X, X+1, and so forth
27	Run 170 multiply and encode MFOVF.
	X Start or run Multiply/Divide common with entry point equal to 0; U0 through U6, U8; and Multiply/Divide start.
	X+1 Copy Multiply/Divide Output mux common, except Mux Code = 7 with entry point equal to 1; Normalize Code = 3 (encode MFOVF); Shift Count Port = 1 (shift according to significance count); and Multiply/Divide run.
28	Run 180 multiply and encode MFOVF, disable MPY80.
	X Start or run Multiply/Divide common with entry point equal to 10(HEX); U0 through U6; and Multiply/Divide start.
	X+1 Copy Multiply/Divide Output mux common, except Mux Code = 7 with entry point equal to 1; Normalize Code = 3 (encode MFOVF); Shift Count Port = 1 (shift according to significance count); and Multiply/Divide run.
29	Run 170 multiply and encode MFOVF, disable MPY70.
	X Start or run Multiply/Divide common with entry point equal to 10(HEX); U0 through U6, U8; and Multiply/Divide start.
	X+1 Copy Multiply/Divide Output Mux common, except Mux Code = 7 with entry point equal to 1; Normalize Code = 3 (encode MFOVF); Shift Count Port = 1 (shift according to significance count); and Multiply/Divide run.
30	Select Multiply/Divide sign from bit 0.
	X Start or run Multiply/Divide common, except transfer RF09 to ADATB and ADATC with entry point equal to 10(HEX); U0 through U6 and Multiply/Divide start.
	X+1 Start or run Multiply/Divide common with entry point equal to 10(HEX) and Multiply/Divide run.
	X+2 Copy Multiply/Divide Output mux common; except Mux Code = 8 and SKG Code = 9, enable endcase with Endcase Code equal to Hold Previous Condition, with entry point equal to 10 (HEX); and Multiply/Divide run.
31	Disable select Multiply/Divide sign from bit 0.
	X Start or run Multiply/Divide common, except transfer RF09 to ADATB and ADATC with entry point equal to 10(HEX); U0, U2, U3, U5, U6, U13, and Multiply/Divide start.
	X+1 Start or Run Multiply/Divide common with entry point equal to 10(HEX) and Multiply/Divide Run.
	X+2 Copy Multiply/Divide Output mux common; except Mux Code = 8 and SKG Code = 9; enable endcase with Endcase Code equal to Hold Previous Condition, with entry point equal to 10 (HEX); and Multiply/Divide run.

Microrand Sequence Number	Short Description of Sequence followed by Individual Microrand Description as indicated by X, X+1, and so forth
32	Select Multiply/Divide sign from bit 32.
	X Start or run Multiply/Divide common, except transfer RF09 to ADATB and ADATC with entry point equal to 10(HEX); U0, U3, U6, and Multiply/Divide start.
	X+1 Start or run Multiply/Divide common with entry point equal to 10(HEX) and Multiply/Divide run.
	X+2 Copy Multiply/Divide Output mux common; except Mux Code = 8 and SKG Code = 9; enable endcase with Endcase Code equal to Hold Previous Condition, with entry point equal to 10 (HEX); and Multiply/Divide run.
33	Enter Multiply/Divide sign to General network sign on Multiply/Divide start.
	X Start or run Multiply/Divide common with entry point equal to 10(HEX); U0, U2, U5; Normalize Code = 6; Carry Code = 6; Adder Code = Add; and Multiply/Divide start.
	X+1 Copy Multiply/Divide Output mux common; except Mux Code = 9, and SKG Code = F(HEX); with entry point equal to 10 (HEX); and Multiply/Divide run.
34	Enter Multiply/Divide sign to General network sign on Multiply/Divide run.
	X Start or run Multiply/Divide common; except transfer RF09 to ADATB and ADATC with entry point equal to 10(HEX); U0 through U6; and Multiply/Divide start.
	X+1 Start or run Multiply/Divide common with entry point equal to 10(HEX); U0 through U6; and Multiply/Divide run.
	X+2 Copy Multiply/Divide Output mux common; except Mux Code = 9; and SKG Code = F(HEX) with entry point equal to 10 (HEX); and Multiply/Divide start.

Micrand Sequence Number — Short Description of Sequence followed by Individual Micrand Description as indicated by X, X+1, and so forth

35 Run multiply, extend find sign and output through the Large Adder.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0, U3, U6; and Multiply/Divide run.
- X+1 — Start or run Multiply/Divide common with entry point equal to 1; U0 through U6; and Multiply/Divide run.
- X+2
  - Transfer RF0B to ADATB.
  - Transfer RF09 to ADATC.
  - Issue following ALN Functional Unit micrand:
    - Micrand Constant = 3F(HEX).
    - Mux Code = F(HEX).
    - Extend zeros in shifter.
    - Carry Code = 6.
    - Adder Code = Add.
    - Output Mux Control = Large Adder bits 0 through 47.
    - Multiply/Divide run.
    - Entry point = 11(HEX).
  - Write RF0A.

36 Test Major Cycle Control buffer for holding of U6.

- X — Start or run Multiply/Divide common, except transfer RF0B to ADATC with entry point equal to 0; U6; and Multiply/Divide start.
- X+1 — Start or run Multiply/Divide common with entry point equal to 0 and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 1.

37 Test Major Cycle Control buffer for holding of U4 and U6.

- X — Start or run Multiply/Divide common, except transfer RF0B to ADATC with entry point equal to 0; U4, U6; and Multiply/Divide start.
- X+1 — Start or run Multiply/Divide common with entry point equal to 0 and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 1.

38 Test Major Cycle Control buffer for holding of U8.

- X — Start or run Multiply/Divide common, except transfer RF0B to ADATC with entry point equal to 0; U5, U6, U8; and Multiply/Divide start.
- X+1 — Start or run Multiply/Divide common with entry point equal to 0 and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 1.

Micrand Sequence Number — Short Description of Sequence followed by Individual Micrand Description as indicated by X, X+1, and so forth

39 Test Major Cycle Control buffer for holding of U3.

- X — Start or run Multiply/Divide common, except transfer RF09 to ADATB and ADATC with entry point equal to 0; U0, U3 through U6; and Multiply/Divide start.
- X+1 — Start or run Multiply/Divide common with entry point equal to 0 and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 1.

40 Test Major Cycle Control buffer for holding of U2.

- X — Start or run Multiply/Divide common, except transfer RF09 to ADATB and ADATC with entry point equal to 0; U0, U2 through U6 and Multiply/Divide start.
- X+1 — Start or run Multiply/Divide common with entry point equal to 0 and Multiply/Divide start.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 1.

41 Test Major Cycle Control buffer for holding of U1.

- X — Start or run Multiply/Divide common, except transfer RF09 to ADATB and ADATC with entry point equal to 0; U0 through U6; and Multiply/Divide start.
- X+1 — Start or run Multiply/Divide common with entry point equal to 0 and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 1.

42 Start and output divide unsigned.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0, U2, U3, U5, U6, U13; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1.

43 Start and output divide signed.

- X — Start or run Multiply/Divide common with entry point equal to 11(HEX); U0 through U6; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1.

44 Start and output divide, 2s complement integer.

- X — Start or run Multiply/Divide common with entry point equal to 11(HEX); U0 through U7; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1.

Micrand Sequence Number | Short Description of Sequence followed by Individual Micrand Description as indicated by X, X+1, and so forth

45 Enable 64-bit integer divide fault and overflow.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0 through U7; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common; except SKG Code = F(HEX), with U0 through U3; and Mux Code = F(HEX), with U4 through U7, Error Code = 6 (Enable Divide Fault); entry point equal to 10(HEX), and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with Error Code = 3 (Enable Divide Overflow); entry point equal to 10(HEX); Multiply/Divide run; Clock Condition register; and Set PONR.

46 Enable 32-bit integer divide fault and overflow.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0, U3, U6, U7; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common; except SKG Code = 9, with U0, U3; Mux Code = F(HEX), with U6, U7; entry point equal to 10(HEX); and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 10(HEX); Multiply/Divide Run Clock Condition register, and Set PONR.

47 Enable 48-bit divide fault and overflow.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0, U2, U3, U5, U6 and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common; except SKG Code = B(HEX), with U0, U2, U3; Mux Code = F(HEX), with entry point equal to 10(HEX); U5, U6; and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 10(HEX), Multiply/Divide run Clock Condition register, and Set PONR.

48 Read UCR.

- X — Live register read of the UCR to RFOD.

49 Start and output divide, 48-bit entry, signed.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0, U2, U3, U5, U6; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1, and Multiply/Divide run.

Micrand Sequence Number | Short Description of Sequence followed by Individual Micrand Description as indicated by X, X+1, and so forth

50 Load and output divide, 32-bit signed integer.

- X — Start or run Multiply/Divide common with entry point equal to 11(HEX); U0, U3, U6, U7; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1 and Multiply/Divide run.

51 Load and output divide, 64-bit unsigned integer.

- X — Start or run Multiply/Divide common with entry point equal to 10(HEX); U0, U2, U3, U5, U6, U13; and Multiply/Divide run.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1 and Multiply/Divide run.

52 Start and output divide, 48-bit signed.

- X — Start or run Multiply/Divide common with entry point equal to 10(HEX); U0, U2, U3, U5, U6; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1.

53 Start and output divide, 48-bit signed.

- X — Start or run Multiply/Divide common with entry point equal to 11(HEX); U0, U2, U3, U5, U6; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1 and Multiply/Divide run.

54 Start, run and output divide, 48-bit signed.

- X — Start or run Multiply/Divide common with entry point equal to 11(HEX); U0, U2, U3, U5, U6; and Multiply/Divide start.
- X+1 — Start or run Multiply/Divide common with entry point equal to 2 and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 1.

55 Start, run and output divide, 48-bit signed.

- X — Start or run Multiply/Divide common with entry point equal to 11(HEX); U0, U2, U3, U5, U6; and Multiply/Divide start.
- X+1 — Start or run Multiply/Divide common with entry point equal to 12(HEX) and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 1 and Multiply/Divide run.

Microrand Sequence Number	Short Description of Sequence followed by Individual Microrand Description as indicated by X, X+1, and so forth
56	Test divide round control.
	X — Start or run Multiply/Divide common with entry point equal to 11(HEX); U0, U2, U3, U5, U6; and Multiply/Divide start.
	X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1 and Multiply/Divide run.
57	Load quotient and encode MFOVF.
	X — Start or run Multiply/Divide common with entry point equal to 0; U0 through U7; and Multiply/Divide start.
	X+1 — Copy Multiply/Divide Output mux common; except Mux Code = 7, with entry point equal to 10(HEX); Normalize Code = 3; Shift Count Port Code = 1; SKG Code = 6; and Multiply/Divide run.
58	Test Quotient register transfer through BIT on 64-bit integer.
	X — Start or run Multiply/Divide common with entry point equal to 0; U0 through U7; and Multiply/Divide start.
	X+1 — Start or Run Multiply/Divide common with entry point equal to 1; U0 through U7; and Multiply/Divide run.
	X+2 — Copy Multiply/Divide Output mux common with entry point equal to 2 and Multiply/Divide run.
59	Test Quotient register transfer through BIT on 32-bit integer.
	X — Start or run Multiply/Divide common with entry point equal to 0; U0, U2, U6, U7; and Multiply/Divide start.
	X+1 — Start or run Multiply/Divide common with entry point equal to 1; U0, U2, U6, U7; and Multiply/Divide run.
	X+2 — Copy Multiply/Divide Output mux common with entry point equal to 2 and Multiply/Divide run.
60	Run 64-bit integer divide.
	X — Start or run Multiply/Divide common with entry point equal to 0; U0 through U7; and Multiply/Divide start.
	X+1 — Start or run Multiply/Divide common with entry point equal to 1; U0 through U7; and Multiply/Divide run.
	. . .
	X+16 — Start or run Multiply/Divide common with entry point equal to 1; U0 through U7; and Multiply/Divide run.
	X+17 — Copy Multiply/Divide Output mux common with entry point equal to 2 and Multiply/Divide run.

Microrand Sequence Number	Short Description of Sequence followed by Individual Microrand Description as indicated by X, X+1, and so forth
61	Test Pop Counter
	X — Start or run Multiply/Divide common with entry point equal to 10(HEX); U0, U2, U3, U5, U6, U13; and Multiply/Divide start.
	X+1 — Start or run Multiply/Divide common with entry point equal to 1; U0, U2, U3, U5, U6, U13; and Multiply/Divide run.
	. . .
	X+15 — Start or run Multiply/Divide common with entry point equal to 1; U0, U2, U3, U5, U6, U13; and Multiply/Divide run.
	X+16 — Copy Multiply/Divide Output mux common with entry point equal to 11(HEX), and Multiply/Divide run.
62	Test C register parity test using unsigned entry.
	X — Start or run Multiply/Divide common; except Mux Code = B(HEX), with entry point equal to 0; U0, U2, U3, U5, U6, U13; and Multiply/Divide start.
	X+1 — Copy Multiply/Divide Output mux common with entry point equal to 1 and Multiply/Divide start.
63	Test B register and Multiply Final Adder parity using unsigned entry.
	X — Start or run Multiply/Divide common; except Mux Code = B; with entry point equal to 10(HEX); U0, U2, U3, U5, U6, U13, and Multiply/Divide start.
	X+1 — Copy Multiply/Divide Output mux common with entry point equal to 11(HEX); and Multiply/Divide start.
	X+2 — Copy Multiply/Divide Output mux common; except write RF09 with entry point equal to 12(HEX); and Multiple/Divide start.
64	Test Major Cycle Hold buffer for U7.
	X — Start or run Multiply/Divide common with entry point equal to 10(HEX); U0 through U7; and Multiply/Divide start.
	X+1 — Start or run Multiply/Divide common with entry point equal to 0 and Multiply/Divide run.
	X+2 — Copy Multiply/Divide Output mux common with entry point equal to 1.
65	Local ALN idle state to clear any parity error.
	X — Issue to address control.

Micrand Sequence Number — Short Description of Sequence followed by Individual Micrand Description as indicated by X, X+1, and so forth

66 Test Major Cycle Control buffer for U9.

- X — Start or Run Multiply/Divide common with entry point equal to 11(HEX); U0, U2, U3, U5, U6, U9; and Multiply/Divide start.
- X+1 — Start or run Multiple/Divide common with entry point equal to 11(HEX) and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 1.

67 Run Multiply from ALN soft control entry points 10(HEX), 11(HEX), 12(HEX), and 13(HEX).

- X — Start or run Multiply/Divide common with entry point equal to 10(HEX); U0 through U6, and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 11(HEX).
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 12(HEX).
- X+3 — Copy Multiply/Divide Output mux common, except write RF09 with entry point equal to 13(HEX).

68 Start and output divide unsigned.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0, U2, U3, U5, U6, U13; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 2.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 12(HEX).

69 Start and output divide signed.

- X — Start or run Multiply/Divide common with entry point equal to 11(HEX); U0 through U6; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 2.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 12(HEX).

70 Start and output divide, 2s complement integer.

- X — Start or run Multiply/Divide common with entry point equal to 11(HEX); U0 through U7; and Multiply/Divide start.
- X+1 — Copy Multiply/Divide Output mux common with entry point equal to 2.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 12(HEX).

Micrand Sequence Number — Short Description of Sequence followed by Individual Micrand Description as indicated by X, X+1, and so forth

71 Test Quotient register transfer through BIT on 64-bit integer.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0 through U7; and Multiply/Divide start.
- X+1 — Start or Run Multiply/Divide common with entry point equal to 1; U0 through U7; and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 2 and Multiply/Divide run.
- X+3 — Copy Multiply/Divide Output mux common with entry point equal to 12(HEX) and Multiply/Divide run.

72 Test Quotient register transfer through BIT on 32-bit integer.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0, U2, U6, U7; and Multiply/Divide start.
- X+1 — Start or run Multiply/Divide common with entry point equal to 1; U0, U2, U6, U7; and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 2 and Multiply/Divide run.
- X+3 — Copy Multiply/Divide Output mux common with entry point equal to 12(HEX) and Multiply/Divide run.

73 Run 64-bit integer divide.

- X — Start or run Multiply/Divide common with entry point equal to 0; U0 through U7; and Multiply/Divide start.
- X+1 — Start or run Multiply/Divide common with entry point equal to 1; U0 through U7; and Multiply/Divide run.
- . .
- . .
- . .
- X+16 — Start or run Multiply/Divide common with entry point equal to 1; U0 through U7; and Multiply/Divide run.
- X+17 — Copy Multiply/Divide Output mux common with entry point equal to 2 and Multiply/Divide run.
- X+18 — Copy Multiply/Divide Output mux common with entry point equal to 12(HEX) and Multiply/Divide run.

74 Test Major Cycle Hold buffer for U7.

- X — Start or run Multiply/Divide common with entry point equal to 10(HEX); U0 through U7; and Multiply/Divide start.
- X+1 — Start or run Multiply/Divide common with entry point equal to 11(HEX) and Multiply/Divide run.
- X+2 — Copy Multiply/Divide Output mux common with entry point equal to 2.
- X+3 — Copy Multiply/Divide Output mux common with entry point equal to 12(HEX).

CONTROL DATA

4201 North Lexington Avenue Saint Paul, Minnesota 55126-6198